



**ITS**

Institut  
Teknologi  
Sepuluh Nopember

**TUGAS AKHIR - TJ141502**

**DIORAMA VIRTUAL EKSPLORATIF KEHIDUPAN MANUSIA  
PURBA INDONESIA MENGGUNAKAN MIXED REALITY  
BERBASIS APPLE ARKIT**

Bima Panji Yudasmara  
NRP 07211440000033

Dosen Pembimbing  
Dr. Surya Sumpeno, S.T., M.Sc.  
Dr. Eko Mulyanto Yuniarno, S.T., M.T.

DEPARTEMEN TEKNIK KOMPUTER  
Fakultas Teknologi Elektro  
Institut Teknologi Sepuluh Nopember  
Surabaya 2018

*[Halaman ini sengaja dikosongkan].*





**ITS**

Institut  
Teknologi  
Sepuluh Nopember

**FINAL PROJECT - TJ141502**

**EXPLORABLE VIRTUAL DIORAMA OF INDONESIAN  
PREHISTORIC HUMAN LIFE USING APPLE ARKIT BASED  
MIXED REALITY**

Bima Panji Yudasmara  
NRP 07211440000033

Advisor

Dr. Surya Sumpeno, S.T., M.Sc.

Dr. Eko Mulyanto Yuniarno, S.T., M.T.

Department of Computer Engineering  
Faculty of Electrical Technology  
Sepuluh Nopember Institute of Technology  
Surabaya 2018

*[Halaman ini sengaja dikosongkan].*

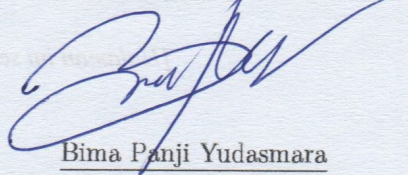
## PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul **"Diorama Virtual Eksploratif Kehidupan Manusia Purba Indonesia Menggunakan Mixed Reality Berbasis Apple ARKit"** adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, Juli 2018



Bima Panji Yudasmara

NRP. 07211440000033

*[Halaman ini sengaja dikosongkan].*



## LEMBAR PENGESAHAN

Diorama *Virtual Eksploratif* Kehidupan Manusia Purba Indonesia  
Menggunakan *Mixed Reality* Berbasis Apple ARKit

Tugas Akhir ini disusun untuk memenuhi salah satu syarat memperoleh gelar  
Sarjana Teknik di Institut Teknologi Sepuluh Nopember Surabaya

Oleh : Bima Panji Yudasmara (NRP: 07211440000033)

Tanggal Ujian : 28 Juni 2018

Periode Wisuda : September 2018

Disetujui oleh:

Dr. Surya Sumpeno, S.T., M.Sc.

NIP: 196906131997021003

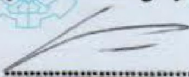
(Pembimbing I)



Dr. Eko Mulyanto Yuniarno, S.T., M.T.

NIP: 196806011995121009


(Pembimbing II)



Mochamad Hariadi, S.T., M.Sc., Ph.D.

NIP: 196912091997031002

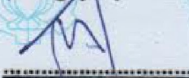
(Penguji I)



Dr. Supeno Mardi Susiki N., S.T., M.T.

NIP: 197003131995121001

(Penguji II)



Ahmad Zaini, S.T., M.Sc.

NIP: 197504192002121003

(Penguji III)



Mengetahui

Kepala Departemen Teknik Komputer

Dr. I Ketut Eddy Purnama, S.T., MT.

NIP. 196907301995121001



*Halaman ini sengaja dikosongkan*

# ABSTRAK

Nama Mahasiswa : Bima Panji Yudasmara  
Judul Tugas Akhir : Diorama *Virtual* Eksploratif Kehidupan  
Manusia Purba Indonesia Menggunakan  
*Mixed Reality* Berbasis Apple ARKit  
Pembimbing : 1. Dr. Surya Sumpeno, ST., M.Sc.  
2. Dr. Eko Mulyanto Yuniarno, ST., MT.

Diorama manusia purba yang ditampilkan di museum-museum arkeologi di Indonesia masih bersifat statis dan tidak eksploratif. Penelitian ini bertujuan untuk merancang aplikasi diorama *virtual* tentang habitat dan kehidupan manusia purba Indonesia yang menerapkan teknologi *mixed reality* (MR) yang berbasis Apple ARKit. Aplikasi ini memfasilitasi pengguna untuk berkeliling di dalam diorama *virtual* dinamis melalui perangkat iOS yang telah mendukung ARKit. Aplikasi ini memiliki kemampuan untuk memindai lingkungan sekitar pengguna secara 3D dengan mengambil data *point cloud* yang diperoleh melalui ARKit. Hasil pemindaian digunakan untuk membangkitkan lingkungan *virtual* yang peletakan objek-objeknya sesuai dengan peletakan dan ketinggian objek-objek di dunia nyata dengan menggunakan *heightmap* dari hasil pemindaian. Aplikasi ini menggunakan konsep "Pintu Ajaib" sehingga pengguna dapat keluar masuk dunia *virtual* melalui sebuah pintu *virtual*. Hasil pengujian menunjukkan bahwa sistem penataan lingkungan *virtual* memiliki tingkat keberhasilan sebesar 89% dengan rata-rata translasi semesta *virtual* sebesar 5,04 cm dan kemungkinan rotasi sebesar 13,33%. Pengujian kuesioner terhadap 30 responden menunjukkan bahwa sebanyak 43,33% responden terbantu dengan penataan ruangan *virtual* yang sesuai dengan dunia nyata dalam pencegahan tabrakan dengan benda-benda di dunia nyata. Sebanyak 36,67% responden setuju manusia purba bergerak yang ditampilkan membantu responden dalam memahami kehidupan manusia purba dan sebanyak 46,67% responden setuju *Mixed Reality* memberikan pengalaman baru dalam pembelajaran kehidupan manusia purba.

Kata Kunci : Diorama *Virtual*, Manusia Purba Indonesia, *Mixed Reality*, Eksploratif

*Halaman ini sengaja dikosongkan*



# ABSTRACT

Name : Bima Panji Yudasmara  
Title : *Explorable Virtual Diorama of Indonesian Prehistoric Human Life using Apple ARKit based Mixed Reality*  
Advisors : 1. Dr. Surya Sumpeno, ST., M.Sc.  
2. Dr. Eko Mulyanto Yuniarno, ST., MT.

*Prehistoric human diorama which are being displayed in archaeology museums in Indonesia are still static and inexplorable. This research aims to design a virtual diorama application about Indonesian prehistoric human habitat and life which applies Apple ARKit based Mixed Reality (MR) technology. Virtual diorama application lets users to explore dynamic virtual diorama from their ARKit supported iOS devices. Virtual diorama application has an ability to 3D scan the environment around user by taking point cloud data through ARKit. Scanning result is used to generate virtual environment which its objects are arranged according with real-world object arrangement by using heightmap from scanning result. This application uses "Magic Door" concept which the user can get in and out virtual diorama through a virtual door. Test results show that virtual environment arranger system has a success rate of 89% with average global translation of 5,04 cm and global rotation chance of 13,33%. Questionnaire testing to 30 persons showed that 43,33% of respondents were helped by virtual environment arrangements which were matched with real world environment arrangement in prevention of collisions with objects in the real world. There are 36,67% of respondents agree that the displayed dynamic prehistoric human helps respondents in understanding prehistoric human life and 46,67% of the respondents agree that Mixed Reality provides a new experience in learning prehistoric human life.*

*Keywords : Virtual Diorama, Indonesian Prehistoric Human, Mixed Reality, Explorable*

*Halaman ini sengaja dikosongkan*

# KATA PENGANTAR

Puji dan syukur kehadiran Allah SWT atas segala limpahan berkah, rahmat, serta hidayah-Nya, penulis dapat menyelesaikan penelitian ini dengan judul **Diorama *Virtual* Eksploratif Kehidupan Manusia Purba Indonesia Menggunakan *Mixed Reality* Berbasis Apple ARKit**.

Penelitian ini disusun dalam rangka pemenuhan bidang riset di Departemen Teknik Komputer ITS serta digunakan sebagai persyaratan menyelesaikan pendidikan S1. Penelitian ini dapat terselesaikan tidak lepas dari bantuan berbagai pihak. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Keluarga, Ibu, dan Ayah yang telah memberikan dorongan spiritual dan material dalam penyelesaian buku penelitian ini.
2. Kepala Departemen Teknik Komputer ITS, Dr. I Ketut Eddy Purnama, S.T., M.T.
3. Dr. Surya Sumpeno, S.T., M.Sc. dan Dr. Eko Mulyanto Yuniarno, S.T., M.T. atas bimbingan selama mengerjakan penelitian.
4. Bapak-ibu dosen pengajar Bidang Departemen Teknik Komputer, atas pengajaran, bimbingan, dan perhatian yang diberikan kepada penulis selama ini.
5. Seluruh teman-teman *B201-crew* Laboratorium Teknik Komputer dan Telematika.
6. Seluruh partisipan dan teman-teman yang telah membantu dalam pengujian.

Kesempurnaan hanya milik Allah SWT, untuk itu penulis memohon segenap kritik dan saran yang membangun. Semoga penelitian ini dapat memberikan manfaat bagi kita semua. Amin.

Surabaya, Juli 2018

Penulis

*Halaman ini sengaja dikosongkan*

# DAFTAR ISI

Abstrak	i
Abstract	iii
KATA PENGANTAR	v
DAFTAR ISI	vii
DAFTAR GAMBAR	ix
DAFTAR TABEL	xi
<b>1 PENDAHULUAN</b>	<b>1</b>
1.1 Latar belakang . . . . .	1
1.2 Permasalahan . . . . .	1
1.3 Tujuan . . . . .	1
1.4 Batasan masalah . . . . .	2
1.5 Sistematika Penulisan . . . . .	2
<b>2 TINJAUAN PUSTAKA</b>	<b>5</b>
2.1 Manusia Jawa . . . . .	5
2.1.1 Karakteristik Fisik . . . . .	6
2.1.2 Lingkungan Hidup dan Kehidupan . . . . .	6
2.2 ARKit . . . . .	6
2.2.1 <i>Motion Tracking</i> . . . . .	7
2.2.2 <i>Depth Perception</i> . . . . .	8
2.2.3 <i>Plane Detection</i> . . . . .	9
2.3 Mixed Reality . . . . .	9
2.4 <i>Immersion</i> . . . . .	11
2.5 <i>Point Cloud</i> . . . . .	11
2.6 <i>Voxel Grid</i> . . . . .	12
<b>3 DESAIN DAN IMPLEMENTASI SISTEM</b>	<b>13</b>
3.1 Gambaran Umum . . . . .	13
3.2 Penelitian Terkait . . . . .	15
3.3 <i>Storyboard</i> Penggunaan . . . . .	15

3.4	<i>Storyboard</i> Tampilan Antar Muka Pengguna . . . . .	18
3.5	Desain Diorama <i>Virtual</i> . . . . .	20
3.6	Desain Sistem Eksplorasi . . . . .	21
3.7	Desain Sistem Pemindaian . . . . .	22
3.8	Desain Sistem Penataan . . . . .	23
3.9	Alur Kerja . . . . .	23
3.10	Implementasi Diorama <i>Virtual</i> . . . . .	25
3.11	Implementasi Sistem Eksplorasi . . . . .	27
	3.11.1 Implementasi Pintu <i>Virtual</i> . . . . .	28
	3.11.2 Algoritma Penggantian Lingkungan . . . . .	31
3.12	Implementasi Sistem Pemindaian . . . . .	33
	3.12.1 Format Penyimpanan . . . . .	37
3.13	Implementasi Penataan Objek . . . . .	37
	3.13.1 Metode <i>Downsampling</i> . . . . .	38
	3.13.2 Algoritma <i>Heightmapping</i> . . . . .	40
	3.13.3 Algoritma Peletakan Objek <i>Virtual</i> . . . . .	40
	3.13.4 Peletakan Pintu dan Manusia Purba <i>Virtual</i> . . . . .	44
<b>4</b>	<b>PENGUJIAN DAN ANALISIS</b>	<b>45</b>
4.1	Pengujian Performasi Sistem . . . . .	45
	4.1.1 Pengujian <i>Frame Rate</i> . . . . .	47
	4.1.2 Pengujian Sistem Penataan Lingkungan <i>Virtual</i> . . . . .	51
4.2	Pengujian Ketergunaan . . . . .	55
	4.2.1 Pengujian Efektivitas . . . . .	56
	4.2.2 Pengujian Kuesioner . . . . .	58
<b>5</b>	<b>PENUTUP</b>	<b>63</b>
5.1	Kesimpulan . . . . .	63
5.2	Saran . . . . .	63
	<b>DAFTAR PUSTAKA</b>	<b>65</b>
	<b>LAMPIRAN</b>	<b>67</b>
	<b>Biografi Penulis</b>	<b>81</b>

# DAFTAR GAMBAR

2.1	Bentuk fisik <i>Homo erectus</i> [1]. . . . .	5
2.2	Sistem <i>VIO</i> [3]. . . . .	7
2.3	<i>Sistem kamera stereo pada umumnya</i> . [7] . . . . .	8
2.4	<i>Virtuality Continuum</i> . [10] . . . . .	9
2.5	Korelasi manusia, komputer, dan lingkungan dalam <i>mixed reality</i> [11]. . . . .	10
2.6	Ilustrasi <i>voxel grid</i> pada <i>point cloud</i> [15]. . . . .	12
3.1	Gambaran umum aplikasi. . . . .	13
3.2	Gambaran umum sistem. . . . .	14
3.3	Penggunaan sistem pemindaian . . . . .	16
3.4	Penggunaan sistem eksplorasi . . . . .	17
3.5	<i>Layout</i> tampilan awal aplikasi . . . . .	18
3.6	<i>Layout</i> tampilan pemindaian dan eksplorasi . . . . .	19
3.7	Denah peletakan objek manusia purba dan pintu <i>virtual</i> . . . . .	21
3.8	Relasi ARKit dengan sistem eksplorasi dan pemindaian. . . . .	22
3.9	Ilustrasi penataan berdasarkan tinggi. . . . .	24
3.10	Objek <i>terrain</i> dan <i>skybox</i> . . . . .	25
3.11	Denah <i>terrain</i> kecil. . . . .	26
3.12	Objek agen pengguna. . . . .	27
3.13	Model pintu. . . . .	28
3.14	Lapisan <i>plane</i> dan <i>collider</i> . . . . .	29
3.15	Tampilan <i>reality plane</i> . . . . .	29
3.16	Tampilan <i>virtual plane</i> . . . . .	30
3.17	Ilustrasi arah sorot kamera <i>virtual</i> . . . . .	30
3.18	<i>Flowchart</i> algoritma pengganti nilai <i>cullingmask</i> . . . . .	32
3.19	<i>Flowchart</i> algoritma proses pemindaian . . . . .	34
3.20	<i>Layout</i> tampilan pemindaian . . . . .	35
3.21	Indikator jumlah titik terpindai. . . . .	35
3.22	<i>Flowchart</i> algoritma penyimpanan tiap frame ke $P_{total}$ . . . . .	36
3.23	<i>Flowchart</i> garis besar algoritma penataan objek. . . . .	37
3.24	Proses penataan objek dari data <i>point cloud</i> . . . . .	38
3.25	<i>Flowchart</i> proses <i>downsampling</i> . . . . .	39

3.26	<i>Flowchart</i> proses <i>heightmapping</i> . . . . .	41
3.27	<i>Flowchart</i> proses pemilihan objek yang akan diletakkan pada suatu koordinat. . . . .	42
3.28	Metode penentuan muat tidaknya objek pada suatu koordinat. . . . .	43
3.29	Nilai tiap koordinat di sekitar objek sebelum dan setelah objek diletakkan. . . . .	44
4.1	Penanda posisi kubus. . . . .	46
4.2	Denah Ruang Rendah. . . . .	52
1	Diorama manusia dan lingkungan purba Museum Manusia Purbakala Sangiran . . . . .	67
2	Diorama manusia purba mengolah batu . . . . .	68
3	Pembatas di sekitar diorama agar tidak dimasuki . . . . .	68
4	Tampilan antar muka awal aplikasi . . . . .	69
5	Profiler proses pemindaian. . . . .	69
6	Status <i>script</i> DioramaManager. . . . .	69
7	Profiler proses eksplorasi. . . . .	70
8	Tampilan antar muka pemindaian . . . . .	70
9	Tampilan antar muka pemindaian . . . . .	71
10	Manusia purba <i>virtual</i> di dalam diorama <i>virtual</i> . . . . .	71
11	Foto benda-benda yang digunakan dalam pengujian performasi . . . . .	72
12	Lokasi tes performasi . . . . .	73
13	Profiler proses penataan. . . . .	73
14	Status <i>Script</i> EnvironmentPlotter. . . . .	73
15	Foto lokasi pengujian Museum Sangiran . . . . .	77
16	Foto beberapa partisipan pengujian di Museum Sangiran . . . . .	78



## DAFTAR TABEL

3.1	Daftar <i>layer</i> tiap <i>layermask</i> . . . . .	31
3.2	Daftar jenis objek tiap <i>layer</i> . . . . .	32
3.3	Tabel pengelompokan objek berdasarkan tinggi . . .	43
4.1	Hasil pengujian sistem pemindaian pada area pertama	48
4.2	Hasil pengujian sistem eksplorasi pada area pertama	49
4.3	Hasil pengujian sistem pemindaian lingkungan pada area kedua . . . . .	49
4.4	Hasil pengujian sistem eksplorasi lingkungan pada area kedua . . . . .	50
4.5	Hasil pengujian sistem pemindaian lingkungan pada area ketiga . . . . .	51
4.6	Hasil pengujian sistem eksplorasi lingkungan pada area pertama . . . . .	51
4.7	Hasil pengujian sistem penataan lingkungan pada ruangan rendah . . . . .	53
4.8	Hasil pengujian sistem penataan lingkungan pada ruangan tinggi . . . . .	54
4.9	Hasil pengujian sistem penataan lingkungan pada ruangan campuran . . . . .	54
4.10	Hasil Pengujian <i>Completion Rate</i> . . . . .	56
4.11	Hasil Pengujian Waktu. . . . .	57
4.12	Daftar pernyataan dalam kuisisioner. . . . .	60
4.13	Persentase respon partisipan terhadap pernyataan pada kuesioner . . . . .	61
1	Hasil pengujian sistem penataan lingkungan pada ruangan rendah . . . . .	74
2	Hasil pengujian sistem penataan lingkungan pada ruangan tinggi . . . . .	75
3	Hasil pengujian sistem penataan lingkungan pada ruangan campuran . . . . .	76
4	Rekapitulasi respon partisipan terhadap pernyataan pada kuesioner . . . . .	79

*Halaman ini sengaja dikosongkan*

# **BAB 1**

## **PENDAHULUAN**

### **1.1 Latar belakang**

Museum arkeologi merupakan museum yang mengkhususkan diri untuk memajang artefak arkeologis. Salah satu museum arkeologi yang terdapat di Indonesia adalah Museum Purbakala Sangiran. Dalam museum arkeologi seperti Museum Sangiran, dapat ditemui benda-benda peninggalan kehidupan purbakala seperti fosil dan alat-alat yang digunakan oleh manusia purba. Rekonstruksi dari manusia purba dan aktifitasnya seperti penggunaan alat-alat manusia purba ditampilkan kepada pengunjung melalui diorama. Diorama-diorama yang terdapat dalam museum arkeologi bersifat statis karena hanya berupa patung sehingga pergerakan seperti cara penggunaan alat purbakala tidak dapat disaksikan oleh pengunjung. Selain itu, diorama yang terdapat di museum arkeologi di Indonesia hanya dapat dilihat dari jarak tertentu.

Solusi dari permasalahan ini adalah dengan memfasilitasi masyarakat untuk masuk ke dalam kehidupan manusia purba virtual untuk memungkinkan pengguna untuk bergerak secara leluasa di dalam dunia virtual sehingga pengguna dapat mengamati isi dari diorama secara seksama. Selain itu, dapat memfasilitasi pengguna agar dapat berinteraksi dengan benda-benda virtual yang terdapat di dalam diorama.

### **1.2 Permasalahan**

Diorama yang digunakan oleh museum arkeologi di Indonesia kurang eksploratif karena pengguna tidak dapat masuk ke dalam diorama dan melihat objek dari dekat. Diorama yang terdapat di museum arkeologi di Indonesia juga kurang memberikan informasi tentang pergerakan dan aktifitas manusia purba karena diorama bersifat statis.

### **1.3 Tujuan**

Merancang aplikasi IOS tentang kehidupan manusia purba yang dapat memungkinkan pengguna untuk seolah olah masuk dan berkeliling di dalam diorama dinamis virtual menggunakan Apple AR-Kit. Aplikasi ini menata diorama secara otomatis berdasarkan ling-

kungan pengguna untuk menunjukkan *obstacle* di sekitar pengguna dan memberikan pengalaman yang berbeda setiap kali pengguna bereksplorasi di ruangan yang berbeda.

## 1.4 Batasan masalah

Pembatasan masalah dilakukan untuk memfokuskan permasalahan yang akan diangkat. Batasan-batasan masalah tersebut diantaranya adalah:

1. Materi yang akan digunakan dalam Diorama *Virtual* Eksploratif Kehidupan Manusia Purba Indonesia Menggunakan *Mixed Reality* Berbasis Apple ARKit adalah reka imajinasi pergerakan dari diorama manusia purba di museum sangiran.
2. Objek 3D dibuat dengan *3D modelling* secara manual.
3. Aplikasi ini tidak dapat membaca pergerakan dan gestur tangan dari pengguna untuk berinteraksi dengan objek dunia *virtual*.
4. Batas antara dunia *virtual* dan nyata akan menggunakan sebuah pintu *virtual* atau sejenisnya.
5. Posisi objek-objek *virtual* yang ditampilkan tidak dapat mengikuti benda di dunia nyata apabila benda di dunia nyata dipindah.
6. Aplikasi ini membutuhkan pencahayaan ruangan yang baik.

## 1.5 Sistematika Penulisan

Laporan penelitian Tugas akhir ini tersusun dalam sistematika dan terstruktur sehingga mudah dipahami dan dipelajari oleh pembaca maupun seseorang yang ingin melanjutkan penelitian ini. Alur sistematika penulisan laporan penelitian ini yaitu :

1. BAB I Pendahuluan  
Bab ini berisi uraian tentang latar belakang permasalahan, penegasan dan alasan pemilihan judul, sistematika laporan, tujuan dan metodologi penelitian.
2. BAB II Tinjauan Pustaka  
Pada bab ini berisi tentang uraian secara sistematis teori-teori yang berhubungan dengan permasalahan yang dibahas pada penelitian ini. Teori-teori ini digunakan sebagai dasar dalam

penelitian, yaitu informasi terkait lingkungan dan kehidupan manusia purba di Museum Manusia Purba Sangiran.

3. BAB III Desain dan Implementasi Sistem

Bab ini berisi tentang penjelasan-penjelasan terkait perencanaan sistem yang akan dilakukan dan hasil implementasi dari rancangan aplikasi.

4. BAB IV Pengujian dan Analisis

Bab ini menjelaskan tentang pengujian penelitian yang sudah dilakukan terhadap aplikasinya. Setiap fitur akan ditunjukkan hasilnya pada bab ini dan dilakukan analisa.

5. BAB V Penutup

Bab ini merupakan penutup yang berisi kesimpulan yang diambil dari penelitian dan pengujian yang telah dilakukan. Penelitian lanjutan merupakan saran dan kritik yang membangun untuk pengembangan lebih lanjut juga ditulis pada bab ini.

*Halaman ini sengaja dikosongkan*

## BAB 2

# TINJAUAN PUSTAKA

Demi mendukung penelitian ini, dibutuhkan beberapa teori penunjang sebagai bahan acuan dan referensi. Dengan demikian penelitian ini menjadi lebih terarah.

### 2.1 Manusia Jawa

Manusia Jawa (*Homo erectus erectus*) adalah jenis manusia purba asal Pulau Jawa yang fosil-fosilnya ditemukan pada tahun 1891 dan 1892. Fosil-fosil Manusia Jawa diekavasi oleh tim yang dipimpin oleh Eugene Dubois di pingiran sungai Bengawan Solo di daerah Trinil, Jawa Timur. Dubois memberi nama species ini *Pithecanthropus Erectus*.



**Gambar 2.1:** Bentuk fisik *Homo erectus*[1].

Manusia Jawa memiliki beberapa kemiripan dengan Manusia Peking (*Sinanthropus pekinensis*) yang membuat kedua spesies tersebut dinamakan ulang menjadi *Homo erectus* oleh Ernst Mayr pada tahun 1950 dan kemudian Manusia Jawa diberikan subspecies menjadi *Homo erectus erectus* pada tahun 1970. Selain di Trinil, Fosil Manusia Jawa juga ditemukan di Sangiran dan Mojokerto.

### 2.1.1 Karakteristik Fisik

Tinggi Manusia Jawa sekitar 173cm dan tulang pinggulnya menunjukkan bahwa mereka berjalan tegak layaknya manusia moderen seperti yang ditunjukkan Gambar 2.1. Tulang pahanya lebih tebal dari manusia moderen. Karakter tengkoraknya bertulang tebal dengan dahi landai dan tak berdagu dan juga memiliki tulang alis yang tajam dan rahang besar. Volume otaknya 900 cc. Manusia Jawa memiliki susunan gigi yang sama dengan manusia dengan gigi taring yang besar. Makanan mereka terdiri dari daging vertebrata, ditinjau dari aspek anatomis dan arkeologis, begitu juga peran ekologis Manusia Jawa. Juga ditemukan bukti bahwa Manusia Jawa menggunakan cangkang kerang untuk memotong daging.

### 2.1.2 Lingkungan Hidup dan Kehidupan

Ada beberapa bukti yang menyatakan bahwa *Homo erectus* hidup di habitat hutan basah. Lebih spesifiknya seperti sebuah sabana yang basah ("*hydroporphic savanna*"). Tanaman yang ditemukan di lokasi eskavasi di trinil antara lain rumput, pakis, tanaman sejenis karet, dan indigofera yang merupakan tanaman umum di daerah hutan hujan[2].

Kayu terbakar ditemukan pada lapisan tanah fosil Manusia Jawa ditemukan di Trinil, yang berusia sekitar 500.000 sampai dengan 830.000 BP menunjukkan bahwa Manusia Jawa menggunakan api. Namun, karena daerah Jawa Tengah merupakan daerah vulkanik, pembentukan batu bara dapat disebabkan oleh api yang disebabkan oleh alam, dan belum ada bukti kuat yang menunjukkan bahwa *Homo erectus* menguasai penggunaan api[1].

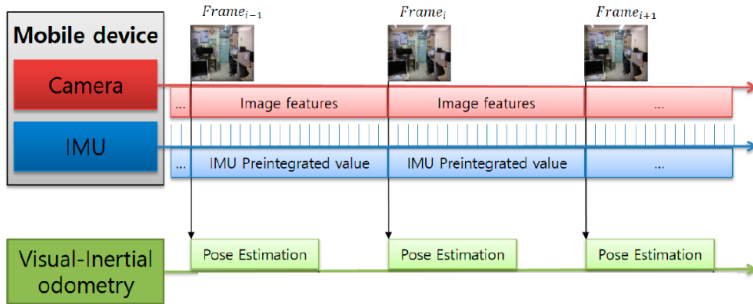
## 2.2 ARKit

ARKit adalah kumpulan *software development tool* yang dikembangkan oleh Apple untuk memudahkan para pengembang aplikasi untuk mengembangkan aplikasi *Augmented Reality* dan *Mixed*



*Reality* karena ARKit memberikan kemampuan seperti *motion tracking* dengan enam DoF dan pendeteksian permukaan. ARKit memerlukan perangkat yang telah dilengkapi prosesor A9 atau yang lebih baru[4]. Kemampuan yang disediakan oleh ARKit untuk digunakan oleh para pengembang adalah sebagai berikut.

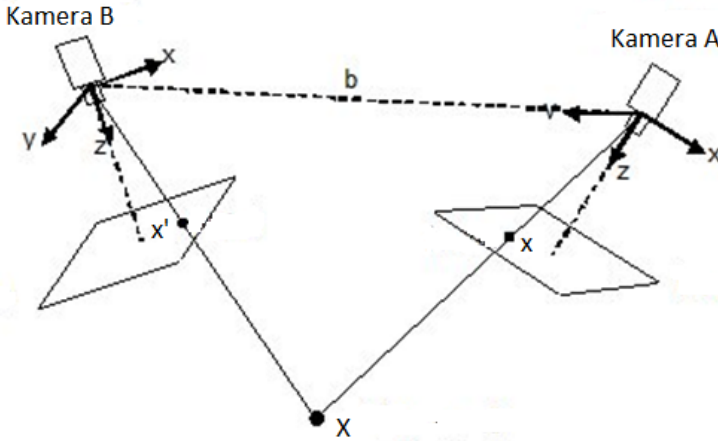
### 2.2.1 Motion Tracking



Gambar 2.2: Sistem VIO[3].

ARKit dapat melakukan *motion tracking* karena ARKit merupakan sebuah sistem *visual inertial odometry* (VIO)[5]. Pada Gambar [6] ditunjukkan bahwa *visual inertial odometry* mengestimasi perubahan posisi dan orientasi (pose) dari sebuah perangkat *mobile* sepanjang waktu menggunakan pengukuran visual dan inersial. Pengukuran visual dilakukan dengan mencocokkan titik-titik fitur pada citra yang ditangkap kamera pada dua buah *frame* dan pengukuran inersial dengan menggunakan sensor-sensor *Inertial Measurement Unit* (IMU) seperti *gyroscope* dan *accelerometer*[6].

Sistem VIO menggunakan sistem visual dan inersial karena pada pergerakan perangkat yang cepat, gambar yang tertangkap kamera akan buram (*blur*) sehingga titik fitur dari gambar tidak bisa didapatkan sehingga sistem inersial dibutuhkan. Di sisi lain, sistem inersial membutuhkan pergerakan yang cukup cepat untuk dapat melacak dengan baik. Pada saat pergerakan perangkat minim, maka digunakan sistem visual yang dapat melacak pergerakan-pergerakan halus [3].



**Gambar 2.3:** Sistem kamera stereo pada umumnya.[7]

### 2.2.2 Depth Perception

*Depth perception* pada ARKit menggunakan satu kamera untuk melakukan proses triangulasi. Pada Gambar 2.3 ditunjukkan dua buah kamera dan sebuah fitur objek yang akan dicari kedalamannya. Titik  $\mathbf{X}$  merupakan titik fitur objek yang akan dicari kedalamannya dalam ruang tiga dimensi dan  $\mathbf{x}$  adalah posisi fitur  $\mathbf{X}$  dalam gambar yang ditangkap kamera A dan  $\mathbf{x}'$  adalah posisi  $\mathbf{X}$  pada gambar yang ditangkap kamera B. Perangkat ARKit yang digunakan hanya menggunakan sebuah kamera, maka dari itu kamera A dan B merupakan pose kamera perangkat pada *frame* A dan B. Untuk mendapatkan posisi  $\mathbf{X}$  dapat menggunakan persamaan matriks proyeksi seperti Persamaan 2.1[8].

$$\mathbf{x} = P\mathbf{X}; \mathbf{x}' = P'\mathbf{X} \quad (2.1)$$

Matriks  $P$  adalah matriks proyeksi. ARKit menggunakan sensor IMU untuk mendapatkan posisi pusat semesta terhadap kamera ( $T$ ) dan rotasi kamera ( $R$ ) untuk digunakan dalam matriks proyeksi

pada Persamaan 2.2. Kedua *frame* menggunakan matriks kalibrasi kamera ( $C$ ) yang sama karena menggunakan satu kamera.

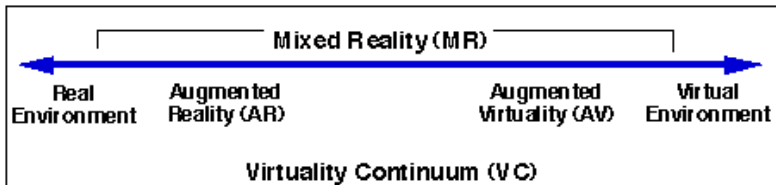
$$P = C[RT] \quad (2.2)$$

Kedalaman tiap fitur pada satu *frame* disajikan dalam bentuk *point cloud* yang dapat diakses oleh pengembang ARKit melalui ARKit SDK.

### 2.2.3 Plane Detection

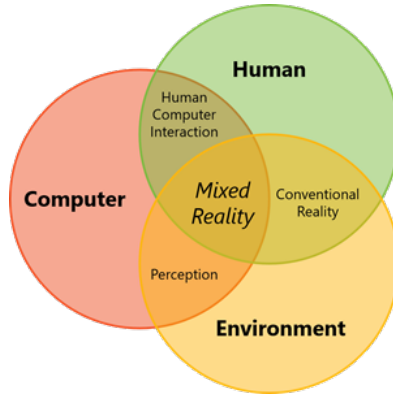
Cara kerja dari pendeteksian permukaan horizontal milik ARKit adalah dengan mencari tiga titik dalam *point cloud* yang berada dalam ketinggian sama (sejajar sumbu  $y$ -nya). Pendeteksian permukaan vertikal dilakukan dengan mengambil tiga titik yang segaris pada sumbu  $xz$ [9].

## 2.3 Mixed Reality



Gambar 2.4: *Virtuality Continuum*. [10]

*Virtuality continuum* merupakan sebuah skala kontinu yang terbentang antara dunia nyata hingga dunia yang seluruhnya *virtual*. *Virtuality continuum* mencakup seluruh kombinasi yang dapat terbentuk antara dunia nyata dan *virtual* seperti *augmented reality* (AR). Area di dalam *virtuality continuum* antara kedua titik ekstrem yang ditunjukkan pada Gambar 2.4 disebut *mixed reality* (MR atau *hybrid reality*). *Mixed reality* merupakan relasi dari *virtual reality* dan *augmented reality*. Dan seiring perkembangan teknologi, Pengaplikasian *mixed reality* tidak hanya dilakukan dengan menggunakan tampilan di layar, tetapi juga menggunakan *input* lingkungan, *spatial sound* dan pembacaan lokasi.



**Gambar 2.5:** Korelasi manusia, komputer, dan lingkungan dalam *mixed reality*[11].

Seperti yang Gambar 2.5 tunjukkan, ada tiga komponen pembentuk *mixed reality*. Komponen pertama adalah komputer, kemudian manusia dan yang terakhir adalah lingkungan. Interaksi yang paling umum adalah interaksi antara manusia dan lingkungannya yang terjadi dalam realita biasa. Interaksi antara manusia dan komputer (HCI) terjadi saat manusia memberikan masukan kepada komputer atau sebaliknya seperti ketika seseorang mengetik menggunakan keyboard atau layar komputer menampilkan luaran melalui layar monitor. Interaksi antara komputer dan lingkungan disebut *environment understanding* atau *environment perception*. *Input* lingkungan dapat memberikan informasi seperti posisi seseorang di dalam ruangan, permukaan dan batas-batas benda, pembacaan cahaya di lingkungan, dan pengenalan objek [11].

Kunci perbedaan antara *mixed reality* dan *augmented reality* terletak pada persepsi lingkungan. *Augmented reality* menampilkan objek *virtual* untuk memperkaya dunia nyata. *Mixed reality* menambahkan persepsi lingkungan untuk menampilkan objek *virtual* yang dapat berinteraksi dengan lingkungan dunia nyata secara *realtime* seperti meletakkan objek dalam sebuah permukaan layaknya bagaimana objek nyata yang dapat berada di posisi yang sama walaupun pengguna bergerak bebas (*anchored*)[10].

## 2.4 Immersion

*Immersion* ke dalam *virtual reality* adalah persepsi sedang berada di dalam dunia yang nyata walau sesungguhnya tak nyata. Persepsi tersebut dibentuk dengan mengelilingi pengguna dengan sistem *virtual reality* yang berupa gambar, suara atau stimulus lain yang menyajikan lingkungan yang menarik secara penuh. *Immersion* dapat dibagi menjadi empat kategori.

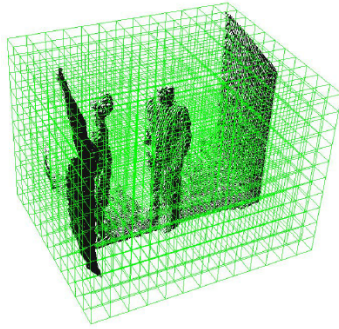
1. ***Tactical Immersion:*** *Tactical immersion* dapat dirasakan saat melakukan operasi taktis yang melibatkan kemampuan pengguna.
2. ***Strategic Immersion:*** *Strategic immersion* lebih merujuk ke otak, dan berhubungan dengan tantangan mental.
3. ***Narrative Immersion:*** *Narrative immersion* terjadi ketika pengguna tertarik ke cerita, serupa dengan pengalaman yang didapatkan ketika membaca buku atau menonton film [12].
4. ***Spatial Immersion:*** *Spatial immersion* terjadi ketika pengguna merasa jika dunia simulasi yang disajikan meyakinkan secara persepsi. Pengguna merasa berada di dalam dunia tersebut dan dunia tersebut terasa nyata [13].

## 2.5 Point Cloud

*Point cloud* adalah "awan" atau kumpulan titik-titik dalam ruang  $nD$  (biasanya  $n = 3$ ). *Point cloud* dapat dinyatakan dengan Persamaan 2.3.

$$p_i = x_i, y_i, z_i \quad P = p_1, p_2, \dots, p_i, \dots, p_n \quad (2.3)$$

*Point cloud* dapat digunakan untuk merepresentasikan informasi tiga dimensi dari lingkungan seperti ketinggian objek atau ukuran ruangan. Selain data koordinat, tiap titik  $p$  dapat memiliki informasi tambahan seperti warna RGB, nilai intensitas, jarak, hasil segmentasi dan lain-lain [14].



**Gambar 2.6:** Ilustrasi *voxel grid* pada *point cloud*[15].

## 2.6 *Voxel Grid*

Sebuah *voxel* merepresentasikan sebuah nilai pada sebuah grid reguler dalam ruang tiga dimensi. Seperti *pixel* dalam sebuah *bitmap*, *voxel* tidak menyimpan nilai posisinya secara eksplisit. Namun, sistem *render* mengetahui posisi sebuah *voxel* berdasarkan posisinya relatif dengan posisi *voxel* lain.

Metode *Voxel Grid* (VG) adalah sebuah metode *downsampling point cloud* sederhana. *Voxel grid* dilakukan dengan mendefinisikan sebuah kisi *voxel* tiga dimensi pada *point cloud* seperti yang ditunjukkan Gambar 2.6. Kemudian, dalam tiap *voxel*, semua titiknya akan di-*downsample* dengan *centroid*-nya. *Downsample* juga dapat dilakukan dengan titik pusat tiap *voxel*, cara ini lebih cepat dari cara pertama tetapi tidak seakurat cara pertama dalam merepresentasikan permukaan sebuah benda [16].

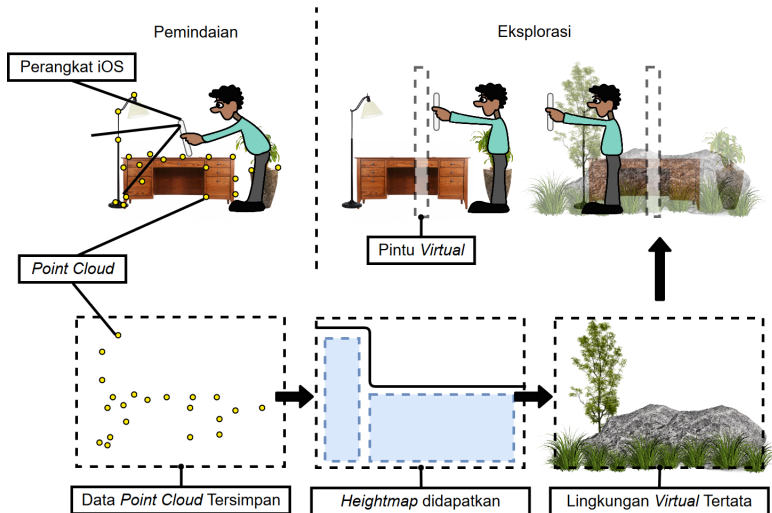
## BAB 3

# DESAIN DAN IMPLEMENTASI SISTEM

Penelitian ini bertujuan untuk membuat aplikasi diorama *mixed reality* tentang manusia purba pulau Jawa berbasis iOS yang dapat memindai ruangan sekitar untuk digunakan sebagai patokan dalam penataan lingkungan diorama *virtual* agar pengalaman tiap pengguna akan berbeda-beda di tiap ruangan dan sekaligus untuk menunjukkan adanya objek atau *obstacle* di sekitar pengguna untuk meningkatkan keamanan saat penggunaan aplikasi.

### 3.1 Gambaran Umum

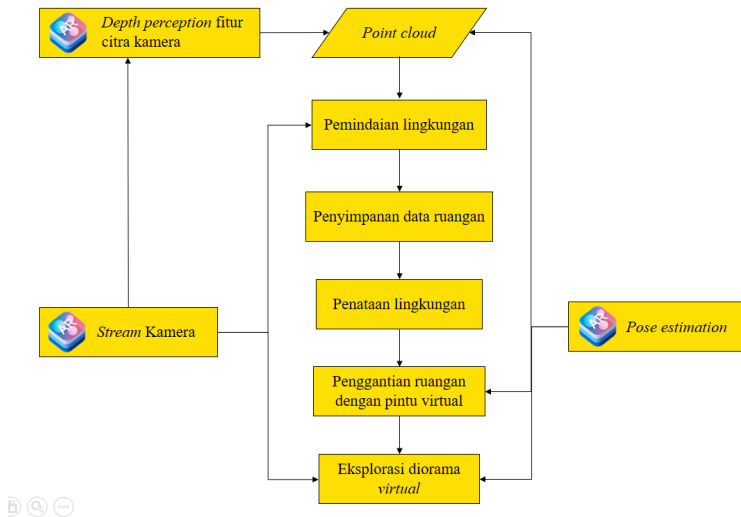
Konsep dasar dari aplikasi ini adalah pintu ke masa manusia purba. Pengguna awalnya disajikan tampilan masa kini dengan *stream* kamera. Kemudian, pengguna dapat seolah-olah masuk ke diorama manusia purba melalui sebuah pintu *virtual*.



**Gambar 3.1:** Gambaran umum aplikasi.

Objek objek yang ditampilkan di dalam diorama *virtual* dipilih dan ditata berdasarkan posisi dan ketinggian objek-objek di dunia

nyata. Pengguna terlebih dahulu memindai ruangan di sekitarnya. Aplikasi kemudian akan menata lingkungan berdasarkan data *point cloud* dari ruangan yang diperoleh seperti yang ditunjukkan Gambar 3.1. Setelah proses penataan selesai, pengguna akan disuguhkan sebuah pintu *virtual* sebagai penghubung antara dunia nyata dan dunia *virtual* yang dapat dimasuki dengan berjalan layaknya memasuki pintu di dunia nyata.



**Gambar 3.2:** Gambaran umum sistem.

Pengembangan aplikasi ini menggunakan library Apple ARKit SDK untuk *depth perception*, pelacakan pose dan *stream* kamera. Pada Gambar 3.2, ditunjukkan bahwa sistem ARKit memberikan data *point cloud* yang merupakan hasil pembacaan kedalaman dari fitur citra yang ditangkap oleh kamera. Data *point cloud* ini kemudian disatukan dengan pose perangkat pada *frame* tersebut dan diterima oleh sistem pemindaian yang mengumpulkan data *point cloud* tiap *frame* untuk disimpan. Pada sistem penyimpanan data *point cloud* disimpan pada sebuah file untuk kemudian dibaca oleh sistem penataan yang membuat *heightmap* dari *point cloud* lingkungan yang tersimpan. Data *heightmap* tersebut kemudian digu-



nakan sebagai acuan dalam peletakan objek-objek di dalam diorama *virtual*. Setelah lingkungan *virtual* tertata, dimunculkanlah pintu *virtual* yang dapat digunakan pengguna untuk keluar masuk dunia *virtual*. Pada sistem eksplorasi dunia *virtual*, digunakan kembali *stream* kamera dan pelacakan pose milik ARKit.

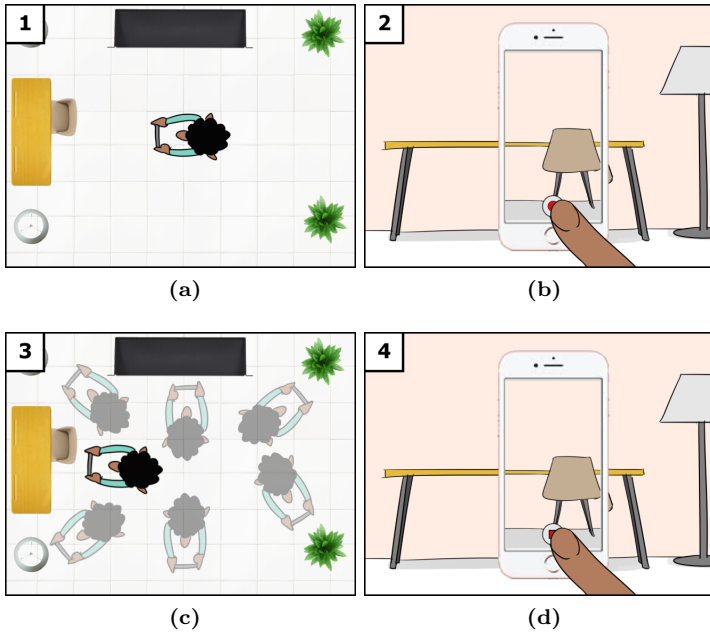
### 3.2 Penelitian Terkait

Penelitian "Eksplorasi Maket *Virtual* Menggunakan *Augmented Reality* dan *Virtual Reality*" oleh Muhammad Aftion membahas sebuah aplikasi yang menerapkan *Augmented Reality* dan *Virtual Reality*[17]. Aplikasi yang dihasilkan penelitian ini menampilkan sebuah maket candi *virtual* yang ditampilkan setelah pengguna memindai marker yang berupa peta Candi Trowulan yang kemudian pengguna dapat berjalan berkeliling untuk melihat maket candi yang ditampilkan menggunakan *headset virtual reality*. Untuk membaca pergerakan penggunaannya, aplikasi tersebut menggunakan *gyroscope* dan *accelerometer*.

Penelitian ini masih belum menerapkan pengenalan lingkungan untuk menjadi *mixed reality* walaupun menerapkan *virtual reality* dan *augmented reality* sehingga memiliki kelemahan dalam penampilan objek *virtual*nya tidak diletakkan tepat di atas tanah. Hal tersebut dapat menyebabkan pengguna merasa bahwa agen pengguna di dalam dunia *virtual* lebih tinggi atau lebih rendah dari tinggi pengguna sebenarnya yang dapat menyebabkan rasa pusing oleh pengguna. Selain itu, pada penelitian tersebut masih memiliki batasan pada sistem eksplorasi yang masih belum dapat memberi tahu pengguna tentang adanya objek di sekitar pengguna.

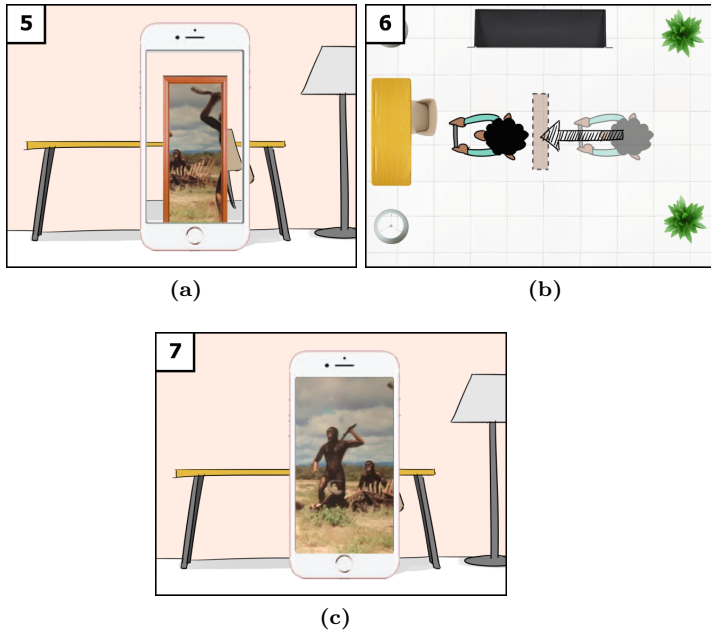
### 3.3 Storyboard Penggunaan

Terdapat dua tahap utama dalam penggunaan aplikasi ini, yaitu tahap pemindaian dan tahap eksplorasi. Pada tahap pemindaian, pengguna memindai lingkungan sekitar pengguna untuk mendapatkan data keadaan fisik di sekitar pengguna. Tahap kedua adalah eksplorasi, pengguna dapat mengeksplorasi dunia *virtual* dari hasil pemindaian di tahap pertama. Pada bagian ini dijelaskan *storyboard* penggunaan aplikasi diorama *virtual*. Berikut adalah *storyboard* penggunaan aplikasi diorama *virtual* ini.



**Gambar 3.3:** Penggunaan sistem pemindaian

1. Pengguna dapat memulai aplikasi di dalam maupun di luar ruangan. Pada Gambar 3.3a, pengguna menggunakan aplikasi di dalam sebuah ruangan. Area yang digunakan harus cukup luas untuk dapat bergerak bebas.
2. Pengguna cukup menekan tombol rekam yang sudah disediakan untuk memulai pemindaian, seperti yang ditunjukkan oleh Gambar 3.3b.
3. Pengguna kemudian memindai ruangan secara menyeluruh dan pastikan tiap sudut ruangan dan tiap objek dalam ruangan terpindai seperti yang ditunjukkan dalam Gambar 3.3c.
4. Pengguna dapat menekan kembali tombol yang sama setelah melakukan pemindaian ruangan, seperti yang diperlihatkan Gambar 3.3d.

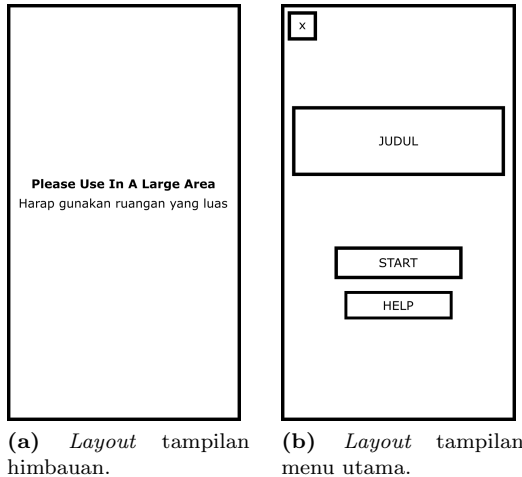


**Gambar 3.4:** Penggunaan sistem eksplorasi

5. Pengguna kemudian akan disajikan tampilan pintu *virtual* di layar perangkat pengguna seperti yang ditunjukkan oleh Gambar 3.4a.
6. Pintu *virtual* yang ditampilkan di layar pengguna dapat berada di posisi yang tetap ketika pengguna berjalan sehingga seperti yang ditunjukkan oleh Gambar 3.4b, pengguna dapat memasuki pintu tersebut hanya dengan berjalan membawa perangkatnya ke arah pintu.
7. Pengguna akan ditampilkan diorama *virtual* di layar perangkat pengguna setelah memasuki pintu *virtual* seperti pada Gambar 3.4c.

### 3.4 *Storyboard* Tampilan Antar Muka Pengguna

Berikut adalah urutan *storyboard* tampilan antar muka pengguna dari diorama *virtual* ini.



**Gambar 3.5:** *Layout* tampilan awal aplikasi

1. Tampilan awal, bagian ini ditampilkan saat pengguna baru menjalankan aplikasi diorama *virtual*. Seperti yang ditunjukkan Gambar 3.5a, pengguna akan diberi sebuah himbauan untuk menggunakan aplikasi diorama *virtual* di ruangan yang luas. Tampilan ini akan menutup secara otomatis setelah tiga detik dan pengguna akan diarahkan langsung ke menu utama.
2. Tampilan Menu Utama, terdapat sebuah tombol "*Play*" yang dapat disentuh untuk berpindah ke tampilan pemindaian. Selain itu, pengguna dapat membuka cara penggunaan dengan menekan tombol "*Help*". *Layout* dari scene ini ditunjukkan pada Gambar 3.5b.
3. Pengguna dapat menekan tombol pemindaian untuk memulai pemindaian lingkungan dalam tampilan pemindaian, seperti yang ditunjukkan Gambar 3.6a



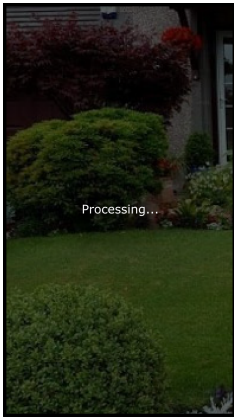
(a) *Layout* tampilan awal pemindaian.



(b) *Layout* tampilan saat pemindaian berjalan.



(c) *Layout* tampilan saat pemindaian dihentikan.



(d) *Layout* tampilan pemrosesan.



(e) *Layout* tampilan pemunculan pintu.



(f) *Layout* tampilan menu utama.

**Gambar 3.6:** *Layout* tampilan pemindaian dan eksplorasi

4. Saat proses pemindaian pengguna dihimbau untuk berkeliling ruangan di sekitar pengguna untuk melakukan pemindaian.

Selama proses pemindaian berlangsung, seperti yang ditampilkan pada Gambar 3.6b, pengguna akan disajikan tampilan *stream* dari kamera belakang dan titik di sekitar objek dalam *stream kamera* untuk menunjukkan bahwa objek tersebut telah terpindai. Setelah pengguna merasa pemindaian telah cukup, maka pengguna dapat menekan tombol *stop* untuk menghentikan pemindaian.

5. Setelah pengguna menghentikan pemindaian akan muncul sebuah notifikasi verifikasi. Pada notifikasi verifikasi pengguna dapat memilih untuk mengakhiri pemindaian sepenuhnya dan menuju ke tampilan eksplorasi. Layout dari notifikasi ini ditunjukkan pada Gambar 3.6c. Selain itu, pengguna juga dapat memilih untuk melanjutkan pemindaian.
6. Pengguna kemudian akan diberi tampilan seperti pada Gambar 3.6d yang memberi tahu bahwa proses pembangkitan lingkungan sedang berlangsung.
7. Setelah proses pembangkitan selesai, seperti yang ditunjukkan Gambar 3.6e, akan dimunculkan sebuah pintu di dunia nyata dalam *stream* kamera dan pengguna dapat berjalan masuk ke dalam pintu tersebut.
8. Saat pengguna memasuki pintu akan disajikan lingkungan diorama *virtual* dan juga manusia purba yang sedang melakukan beberapa aktivitas. Pengguna dapat berjalan mengeksplorasi di dalam lingkungan *virtual* yang disajikan. Pada tampilan ini tidak ada teks maupun tombol menu layaknya di tampilan lain seperti yang ditunjukkan Gambar 3.6f.

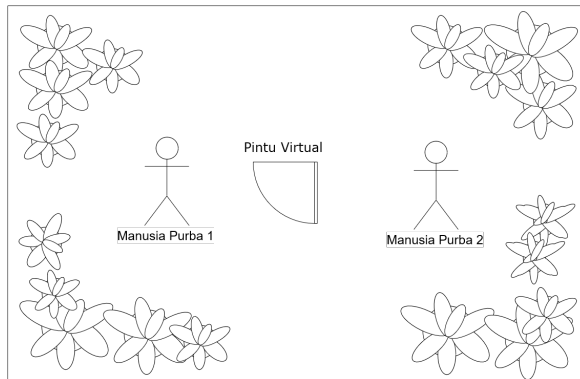
### 3.5 Desain Diorama *Virtual*

Objek-objek lingkungan dibuat dengan merujuk pada lingkungan manusia purba hidup yang ditampilkan dalam diorama yang terdapat dalam museum-museum arkeologi.

Lingkungan diorama yang ditampilkan tidak tertata secara penuh sejak awal. Namun, hanya disiapkan lingkungan dasar (kosong), beberapa aset pohon, rumput, batu, manusia purba dan pintu. Penataan akan dilakukan secara otomatis dengan mengolah data point

cloud yang didapatkan saat pemindaian. Penataan objek-objek lingkungan dilakukan berdasarkan ketinggian objek-objek nyata yang terpindai.

Objek lain yang ditampilkan adalah pintu dan manusia purba *virtual*. Model manusia purba yang digunakan mengacu pada Manusia Jawa. Peletakan pintu dan manusia purba *virtual* ditunjukkan pada Gambar 3.7.



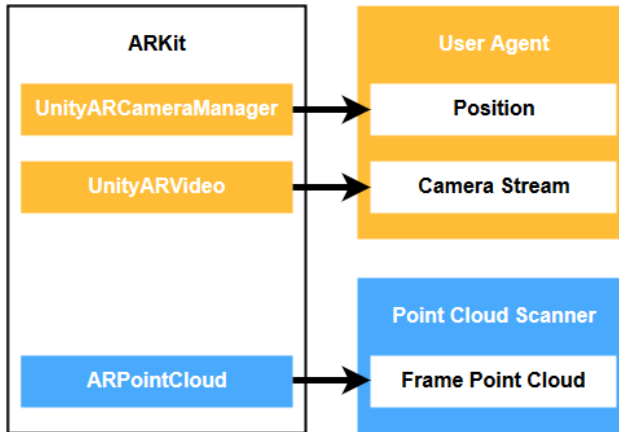
**Gambar 3.7:** Denah peletakan objek manusia purba dan pintu *virtual*.

Pintu *virtual* diletakkan di antara dua manusia purba *virtual* untuk mendorong pengguna untuk berkeliling dan bereksplorasi di dunia *virtual*.

### 3.6 Desain Sistem Eksplorasi

Pada Gambar 3.8 ditunjukkan bahwa sistem eksplorasi ini menggunakan ARKit SDK untuk mengakses *stream* kamera belakang untuk ditampilkan ke pengguna. Selain itu, ARKit SDK juga digunakan untuk mengakses sistem odometri untuk mesinkronasikan perubahan posisi dan rotasi perangkat pengguna di dunia nyata ke dunia *virtual*.

Untuk sistem penggantian tampilan dunia nyata ke dunia *virtual* digunakan objek pintu yang diberikan *collider* sebagai *trigger*-nya. Saat pengguna berjalan masuk ke dalam pintu, *collider* di agen pengguna akan mengaktifkan *trigger* di objek pintu dan akan mengganti dunia yang ditampilkan.



**Gambar 3.8:** Relasi ARKit dengan sistem eksplorasi dan pemindaian.

Penggantian tampilan dunia memanfaatkan *cullingmask* pada kamera Unity. *Cullingmask* digunakan untuk memilih *layer* objek mana saja yang akan *dirender* oleh kamera Unity.

### 3.7 Desain Sistem Pemindaian

Pengguna perlu untuk memindai ruangan tersebut menggunakan kamera perangkatnya untuk mendapatkan data *point cloud* dari ruangan maka. Untuk melakukan pemindaian pengguna perlu untuk berkeliling di dalam ruangan untuk memindai tiap sudut ruangan tersebut. Pengguna akan diperlihatkan stream kamera sekaligus titik-titik *point cloud* yang tertangkap tiap *frame*-nya melalui layar perangkatnya.

Ditunjukkan pada Gambar 3.8, data *point cloud* tiap *frame*-nya dapat diakses melalui variabel *ARPointCloud* dari library ARKit. *ARPointCloud* merupakan *array* dari titik-titik yang dapat diakses posisi globalnya. Data posisi tiap titik ini kemudian dikumpulkan dan disimpan untuk diolah.

Pemindaian ruangan juga mencari koordinat *y* dari *ground*. Koordinat ini didapatkan dengan mencari permukaan terendah yang didapatkah oleh ARKit yang dijadikan sebagai nilai *y* minimum. Pemindaian juga menghitung jumlah titik total dan mencari nilai-nilai



koordinat maksimum dan minimum. Nilai-nilai ini akan digunakan dalam tahap pemetaan.

Apabila pengguna merasa bahwa pemindaian telah cukup maka pengguna dapat menghentikan sementara proses pemindaian. Dalam tahap ini penyimpanan aliran data *point cloud* dihentikan, tetapi variabel tidak dikosongkan dan juga berkas properti belum dibuat. Apabila pengguna memutuskan untuk melanjutkan proses pemindaian maka penyimpanan aliran data *point cloud* dilanjutkan. Sebaliknya, apabila pengguna memutuskan untuk menghentikan pemindaian sepenuhnya, maka variabel-variabel akan disimpan dan berkas properti dibuat, variabel-variabel dalam proses pemindaian juga dikosongkan.

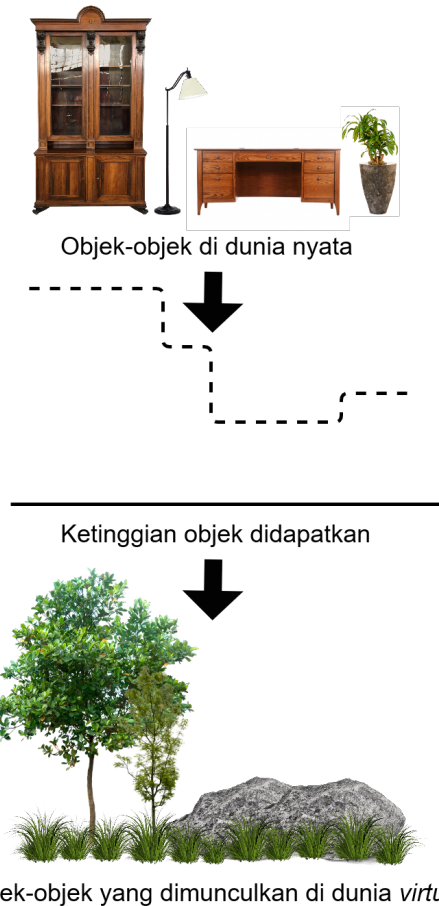
### 3.8 Desain Sistem Penataan

Objek-objek *virtual* ditata berdasarkan data *point cloud* ruangan yang telah dipindai. Data *point cloud* tersebut kemudian akan dijadikan sebuah *heightmap*. *Heightmap* tersebut digunakan untuk menyimpan ketinggian objek-objek yang berada dalam lingkungan nyata. Ketinggian-ketinggian objek itulah yang digunakan untuk menentukan objek apa yang dapat diletakkan dalam suatu posisi di lingkungan *virtual*. Ilustrasi dari penataan objek *virtual* berdasarkan ketinggian objek di dunia nyata ditunjukkan pada Gambar 3.9.

### 3.9 Alur Kerja

Terdapat tujuh tahap utama dalam merancang dan mengimplementasikan desain sistem aplikasi sehingga dapat digunakan sebagai luaran akhir. Berikut merupakan tujuh tahap tersebut :

1. **Studi Manusia Purba.** Mengumpulkan informasi tentang manusia purba yang akan ditampilkan baik kehidupannya, lingkungannya, makanan dan alat-alat yang digunakan. Proses studi termasuk mengunjungi museum manusia purba di Sangiran, Jawa Tengah untuk melihat langsung contoh diorama-diorama yang ditampilkan dan studi melalui beberapa sumber tertulis.
2. **Pengumpulan Aset.** Mencari dan membuat aset-aset yang akan digunakan dalam aplikasi baik aset suara, model 3D ma-



**Gambar 3.9:** Ilustrasi penataan berdasarkan tinggi.

upun gambar 2D. Aset akan dicari dari Assets Store Unity atau dari sumber lain, apabila aset yang tersedia dianggap kurang maka akan dibuat.

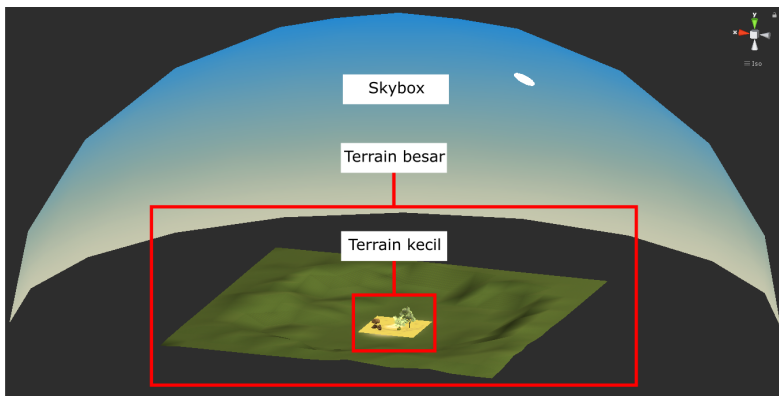
3. **Pembuatan Pintu Antara Dunia Nyata dan *Virtual*.** Pengguna aplikasi dapat berpindah ke dunia virtual dan dunia nyata menggunakan pintu *virtual* yang disediakan. Pintu

dibuat dengan memanfaatkan render texture yang disediakan oleh Unity3D.

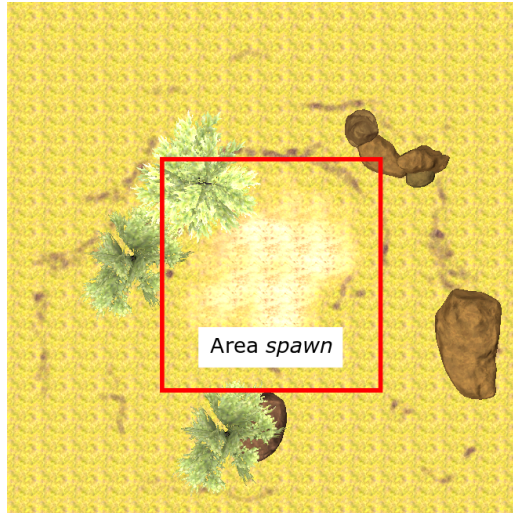
4. **Implementasi Sistem Pemindaian Lingkungan.** Pada tahap ini diterapkan sistem pemindaian lingkungan yang akan menghasilkan data *point cloud* dari lingkungan dunia nyata sekitar pengguna. Data *point cloud* tersebut kemudian disimpan dalam sebuah berkas.
5. **Implementasi Peletakan Objek *Virtual*.** *Point cloud* hasil pemindaian didapatkan *heightmap*-nya sebagai acuan peletakan objek-objek *virtual*, termasuk pintu dan manusia purba, ke dalam lingkungan *virtual*.
6. **Implementasi Interaksi Lingkungan *Virtual*.** Interaksi antara objek-objek virtual dan pengguna untuk membuat pengguna benar-benar berada dalam dunia *virtual*.
7. **Evaluasi.** Interaksi dan visualisasi yang diterapkan dievaluasi dengan pengujian untuk mengukur kesesuaian hasil penelitian dengan hasil yang diharapkan.

### 3.10 Implementasi Diorama *Virtual*

Penataan lingkungan diorama virtual dilakukan secara otomatis setelah melakukan pemindaian. Maka dari itu, hanya disiapkan beberapa aset berupa lahan, manusia purba, pintu, beberapa jenis pohon, batu dan rumput yang disimpan dalam bentuk *prefab unity*.



**Gambar 3.10:** Objek *terrain* dan *skybox*.



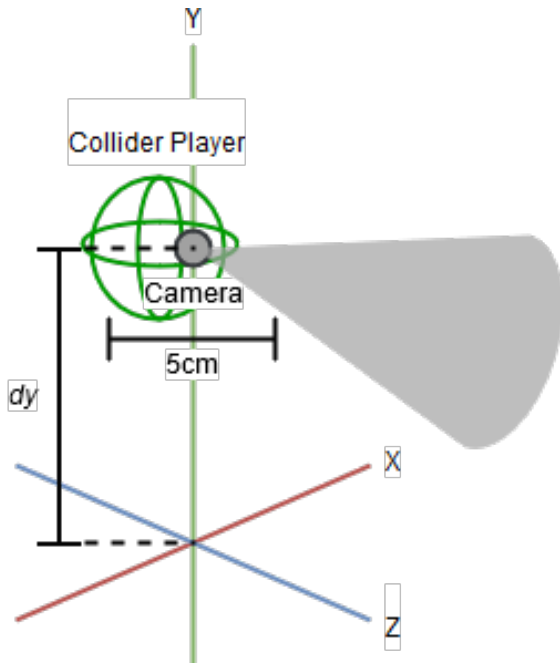
**Gambar 3.11:** Denah *terrain* kecil.

Lahan yang disediakan, seperti yang ditunjukkan Gambar 3.10, terdiri dari dua buah *terrain unity* dan sebuah kubah langit sebagai pengganti *skybox*. *Terrain* pertama adalah *terrain* yang berukuran kecil yang merupakan *terrain* yang akan dieksplorasi oleh pengguna. *Terrain* ini berisi beberapa objek, yakni pepohonan, rerumputan dan bebatuan yang ditata manual di bagian-bagian luar dari *terrain* kecil sebagai dekorasi awal, penataan objek dapat dilihat pada Gambar 3.11. Objek-objek tersebut ditata di bagian luar karena bagian tengah *terrain* akan diisi oleh objek-objek yang dimunculkan secara otomatis dari hasil pemindaian. *Terrain* lainnya adalah *terrain* yang berukuran besar yang digunakan untuk memberikan pemandangan pegunungan yang jauh. Kedua *terrain* ini dipisah untuk memperingan kerja perangkat saat mengolah data *terrain* dalam proses peletakan objek. Kubah langit dibuat karena *skybox* unity akan menghilang ketika *stream* kamera berjalan.

Aset pepohonan, rerumputan dan bebatuan diperoleh dari Unity Assets Store yang dipilih dengan mengacu kepada lingkungan Manusia Jawa yang berupa sabana basah. Aset manusia purba yang digunakan adalah hasil reka imajinasi yang dibuat manual.

### 3.11 Implementasi Sistem Eksplorasi

Disediakan objek kamera untuk membuat pengguna dapat ber-eksplorasi dalam aplikasi ini yang posisi dan rotasinya nya diatur oleh *script* Unity ARCameraManager yang disediakan oleh ARKit SDK.



**Gambar 3.12:** Objek agen pengguna.

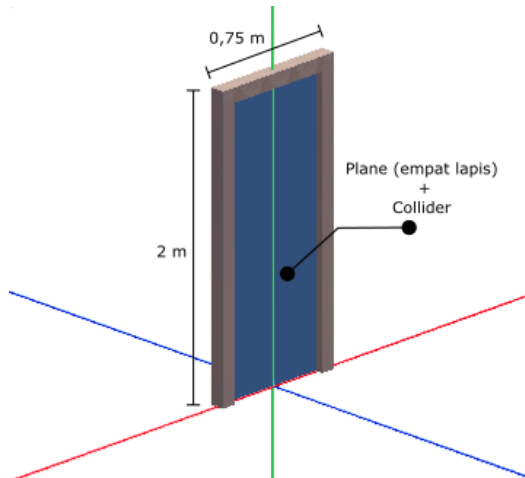
Dalam Gambar 3.12,  $dy$  adalah ketinggian kamera di ruang *virtual* aplikasi. Titik *origin* ( $O$ ) global dari aplikasi ini adalah posisi awal aplikasi ini dijalankan. Oleh karena itu, nilai  $dy$  tidak sama tiap aplikasi dijalankan.

Terdapat pengaturan untuk *clear flag* Pada kamera utama yang dipasang di objek agen pengguna. Pengaturan ini untuk menentukan apa yang akan *render* kamera saat tidak ada objek *virtual* yang tertangkap kamera tersebut. Kamera pada Unity umumnya menggunakan *skybox* sebagai *clear flag*-nya. Pada kamera yang

digunakan sebagai kamera utama dalam aplikasi ini, *clear flags* diatur menjadi "*Depth only*" yang membuat kamera secara *default* tidak *render* apapun saat tidak ada objek *virtual* yang tertangkap. Namun, dengan pengaturan ini UnityARManager dapat mengisi *stream* kamera perangkat untuk *render* sebagai *clear flag* dari kamera utama.

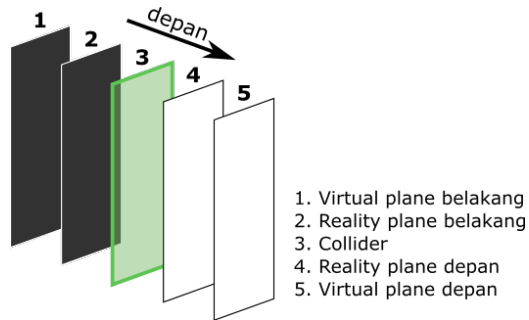
*Collider* dalam agen pengguna digunakan untuk mengganti lingkungan saat agen pengguna melewati pintu. Saat agen pengguna melewati pintu *virtual*, *collider* milik objek pengguna akan mengaktifkan *trigger collider* di pintu untuk mengganti *layermask* di kamera. *Collider* diletakkan persis di belakang kamera karena pengguna dianggap melewati pintu saat kamera tepat menyentuh *collider* di pintu *virtual*.

### 3.11.1 Implementasi Pintu *Virtual*



**Gambar 3.13:** Model pintu.

Bentuk pintu *virtual* yang disajikan adalah berupa bingkai pintu sederhana yang memiliki tinggi 2 m dan lebar 0,75 m. Pada Gambar 3.13, ada empat lapis *plane* dan sebuah *collider* di tengah pintu, untuk isi lapisannya dapat dilihat pada Gambar 3.14.



**Gambar 3.14:** Lapisan *plane* dan *collider*.

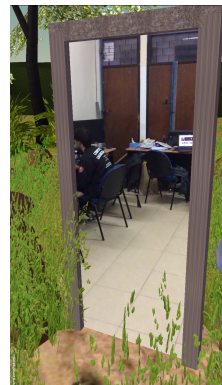
Terdapat dua buah *virtual plane* dan *reality plane*. *Reality plane* digunakan untuk menampilkan dunia nyata saat pengguna berada di dunia *virtual* dan *virtual plane* digunakan untuk menampilkan dunia *virtual* saat pengguna berada di dunia nyata. Kedua jenis *plane* tersebut butuh untuk dibuat dua lapis karena pintu dapat dilihat dari kedua sisi, depan dan belakang. Namun, karena *plane* hanya memiliki satu sisi, maka dibuatlah *plane* kedua yang menghadap ke sisi lainnya.



(a)

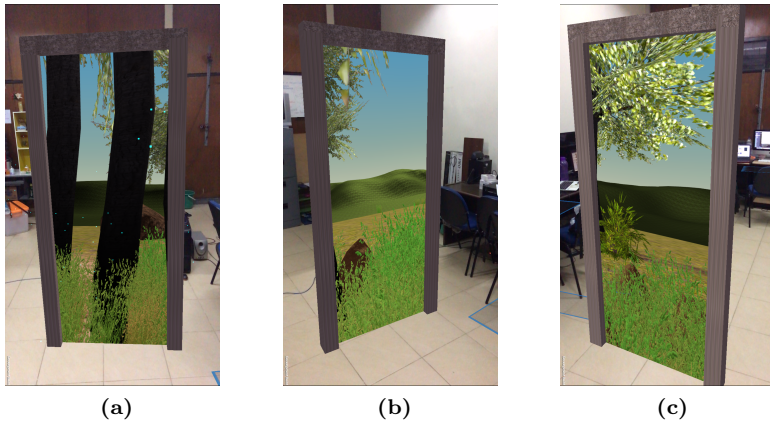


(b)



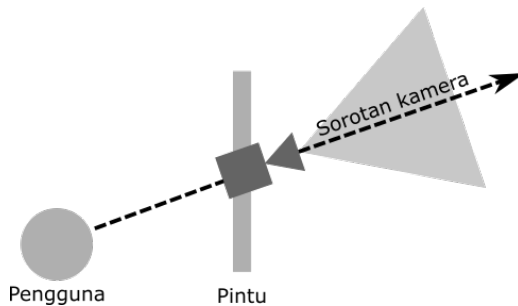
(c)

**Gambar 3.15:** Tampilan *reality plane*



**Gambar 3.16:** Tampilan *virtual plane*

*Reality plane* diberi material yang menggunakan sebuah *shader* yang dibuat secara khusus untuk menampilkan dunia nyata pada *reality plane*. *Shader* ini dirender setelah objek-objek lain di-render, saat objek *reality plane* di-render, *shader* ini akan membuat Unity tidak me-render apa-apa di tiap *pixel* yang seharusnya objek *reality plane* ter-render. Hal ini akan membuat pada bagian yang tak ter-render tersebut hanya menampilkan *clear flag* dari kamera utama yang merupakan *stream* dari kamera perangkat. Gambar 3.15 menunjukkan bagaimana *reality plane* bekerja.



**Gambar 3.17:** Ilustrasi arah sorot kamera *virtual*.



Sebuah kamera yang dipasang di tengah pintu digunakan untuk menampilkan dunia *virtual* pada *virtual plane*. Kamera ini hanya menangkap dunia *virtual* baik saat pengguna berada di dalam dan luar dunia *virtual*. *Virtual plane* diberikan material yang teksturnya menggunakan *render texture* Unity. *Render texture* merupakan sebuah tekstur material yang dapat diisi dengan hasil tangkapan kamera Unity. Hasil tangkapan kamera *virtual* ditampilkan melalui *render texture* pada *virtual plane*. Gambar 3.16 menunjukkan bagaimana *virtual plane* bekerja.

Arah sorotan kamera *virtual* disamakan dengan arah dari posisi pengguna ke kamera agar menampilkan apa yang seharusnya dilihat pengguna saat berada di posisinya dan melihat ke arah pintu untuk membuat pengguna benar-benar merasa sedang melihat ke sebuah pintu ke dunia *virtual*. Ilustrasi arah sorot kamera berdasarkan posisi pengguna ditunjukkan pada Gambar 3.17

### 3.11.2 Algoritma Penggantian Lingkungan

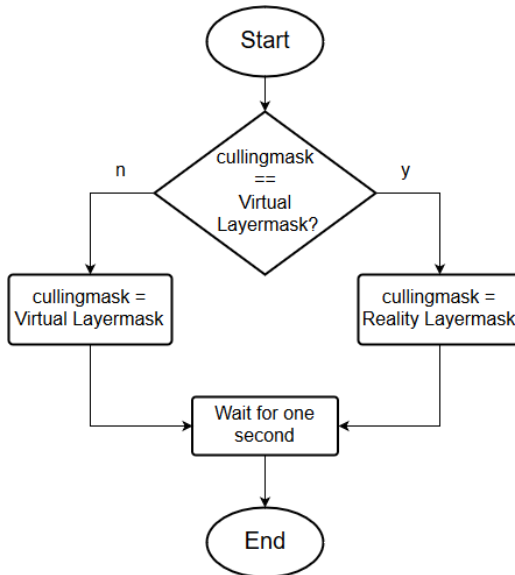
Pilihan layer pada *cullingmask* kamera dapat diisi dengan *layermask*. *Layermask* adalah sebuah variabel yang berisi data tentang layer apa saja yang dipilih. Ada dua *layermask* yang digunakan, yaitu *virtual layermask* dan *reality layermask*. Daftar *layer* dalam kedua *layermask* tersebut dapat dilihat pada Tabel 3.1 dan daftar jenis-jenis objek yang termasuk dalam tiap layer dapat dilihat pada Tabel 3.2. Algoritma pemilihan *layermask* yang diisikan ke *cullingmask* dapat dilihat pada Gambar 3.18.

**Tabel 3.1:** Daftar *layer* tiap *layermask*.

<i>Layermask</i>	<i>Virtual</i>	<i>Realty</i>
<i>Layer</i>	Default TransparentFX Ignore Raycast Water UI Door IgnoreView ARKitPlane VirtualEnvironment RealityGate	Default TransparentFX Ignore Raycast Water UI Door IgnoreView ARKitPlane VirtualGate

**Tabel 3.2:** Daftar jenis objek tiap *layer*.

Jenis Objek	<i>Layer</i>
Pintu	Door
Virtual Plane	Virtual Gate
Reality Plane	Reality Gate
Terrain Skydome Manusia Purba Pohon Batu Rumput	Virtual Environment



**Gambar 3.18:** Flowchart algoritma pengganti nilai *cullingmask*.

Pada Gambar 3.18, waktu interval sebesar satu detik diberikan untuk memastikan pengaktifan *trigger* tidak berjalan secara beruntun secara cepat saat pembacaan posisi pengguna bergeser bolak-balik pada beberapa *frame* yang runtut.

### 3.12 Implementasi Sistem Pemindaian

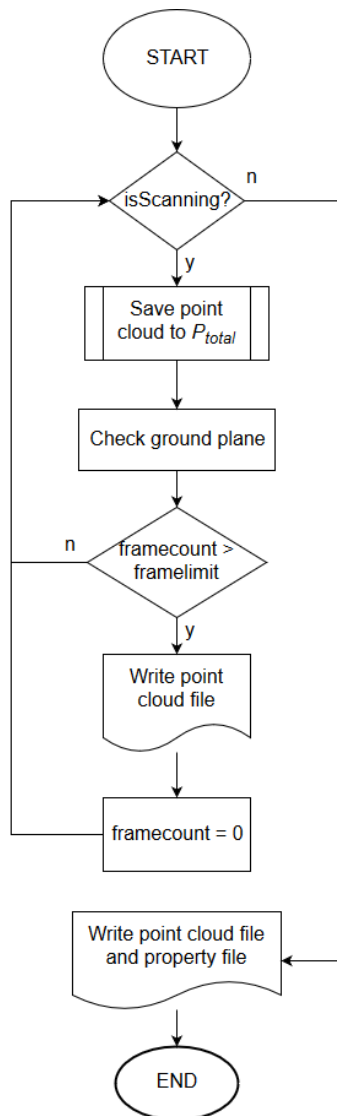
Pengguna berjalan berkeliling ruangan untuk memindai tiap objek dalam ruangan tersebut. Ditampilkan titik-titik di sekitar objek yang terpindai untuk memberi tahu pengguna apabila objek terpindai. Titik-titik tersebut ditampilkan menggunakan Particle-System bawaan Unity. Tampilan titik-titik tersebut dapat dilihat pada Gambar 3.20a.

Alur proses pemindaian digambarkan pada Gambar 3.19. Tiap titik yang ditangkap di tiap *frame* tidak seluruhnya disimpan karena ada *frame* yang *point cloud*-nya sama dengan *frame* sebelumnya karena pengguna tidak berpindah area pemindaian. Apabila *point cloud* dari semua *frame* pemindaian disimpan maka akan membebankan perangkat yang dapat membuat pengguna tidak nyaman dan meningkatkan galat dalam pemindaian seperti titik-titik di sepanjang garis sumbu koordinat. Selain itu, akan menambah ukuran berkas yang disimpan.

*Point cloud* tiap *frame* akan disimpan dalam sebuah variabel penampung ( $P_{total}$ ). Seperti yang ditunjukkan Gambar 3.22, sebelum memasukkan *point cloud* dari ARPointCloud pada tiap *frame* ke variabel penampung, *point cloud frame* tersebut dicek titik pertamanya apakah sama dengan titik pertama *point cloud frame* sebelumnya. Ini dilakukan untuk mengetahui apakah *point cloud frame* tersebut sama dengan *frame* sebelumnya. Apabila sama, maka *point cloud frame* tersebut tidak akan disimpan karena akan memberatkan memori dan proses penyimpanan.

Indikator yang ditunjukkan pada Gambar 3.21 dibuat untuk memastikan pengguna memindai cukup *point cloud*. Indikator diletakkan pada tombol dan akan berubah warna sesuai jumlahnya. Pengguna juga tidak bisa memberhentikan pemindaian sebelum jumlah titik yang terpindai melebihi jumlah minimal.

Pengguna juga harus memindai minimal sebuah *plane* untuk dapat menghentikan pemindaian. Hal ini perlu dilakukan untuk memastikan pengguna mendapatkan sebuah *plane* untuk dijadikan *ground* dari diorama *virtual*. Pengguna diberi himbauan untuk memindai *ground* di dunia nyata agar *ground* di dunia *virtual* sama dengan dunia nyata. Untuk membantu pengguna dalam memahami perintah untuk memindai *ground*, maka himbauan yang ditampilkan



**Gambar 3.19:** *Flowchart* algoritma proses pemindaian

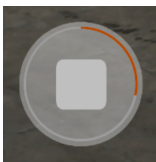


(a) *Layout* tampilan point cloud.

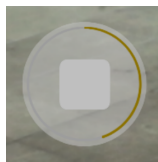


(b) *Layout* tampilan menu utama.

**Gambar 3.20:** *Layout* tampilan pemindaian



(a) Indikator sa-  
at kualitas ren-  
dah.



(b) Indikator sa-  
at kualitas se-  
dang.



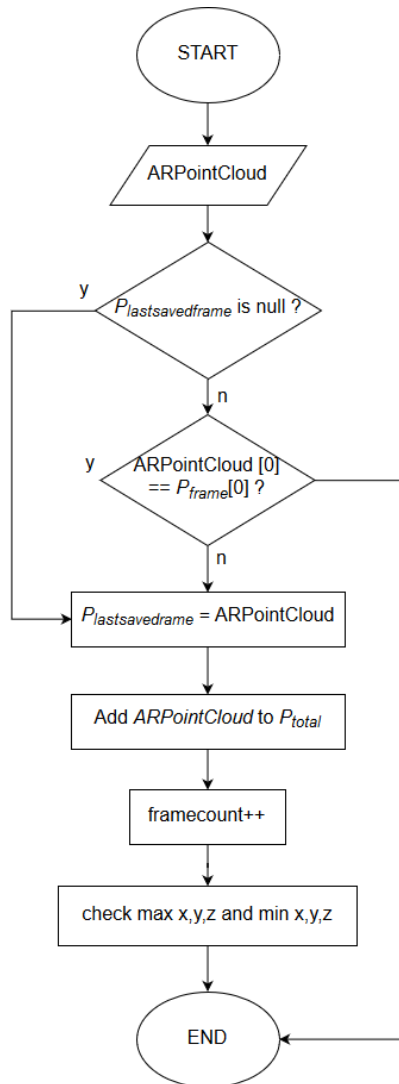
(c) Indikator sa-  
at kualitas ba-  
gus.



(d) Tombol ber-  
henti pindai ak-  
tif.

**Gambar 3.21:** Indikator jumlah titik terpindai.

berbunyi "Pindai lingkungan hingga beberapa bunga tertanam di lantai/tanah" seperti yang ditunjukkan Gambar 3.20a. Untuk mendukung himbauan ini, apabila pengguna menemukan sebuah *plane*,



**Gambar 3.22:** Flowchart algoritma penyimpanan tiap frame ke  $P_{total}$ .

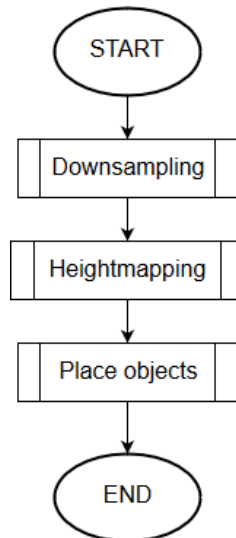
maka akan dimunculkan sebuah bunga *virtual* di *plane* tersebut. Tampilan dari bunga *virtual* ditunjukkan pada Gambar 3.20b.

Variabel penampung *Point cloud* akan disimpan ke sebuah berkas baru setiap beberapa *frame* sekali dan variabel penampung *point cloud* akan dikosongkan. Hal ini dilakukan untuk mengurangi beban RAM karena ukuran variabel penampung menjadi tidak terlalu besar.

### 3.12.1 Format Penyimpanan

Data *point cloud* disimpan menjadi dua jenis berkas, yaitu sekumpulan berkas berisi koordinat-koordinat tiga dimensi dan sebuah berkas properti yang berisi data koordinat maksimum, koordinat minimum, dan jumlah titik total. Data-data properti berformat csv dan disimpan dalam sebuah berkas berekstensi .pcprop.

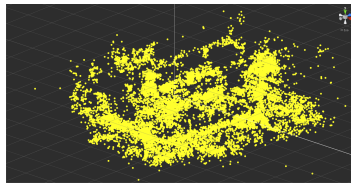
## 3.13 Implementasi Penataan Objek



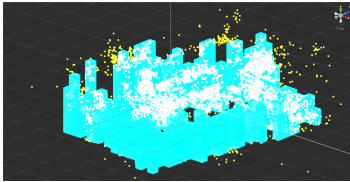
**Gambar 3.23:** Flowchart garis besar algoritma penataan objek.

Data *point cloud* yang diperoleh tidak serta merta dijadikan *heightmap*. Ditunjukkan pada Gambar 3.23, sebelum dijadikan *he-*

*heightmap*, data *point cloud* sebelumnya akan di-*downsampling* agar tidak memberatkan kinerja perangkat mengingat sumber daya yang tersedia sedikit karena perangkat hanya berupa sebuah ponsel cerdas. Pada Gambar 3.24a ditampilkan *point cloud* hasil pemindaian. *Point cloud* tersebut kemudian didapatkan *heightmap*-nya yang ditunjukkan pada Gambar 3.24b dan dari *heightmap* tersebut dimunculkan lingkungannya yang ditampilkan pada Gambar 3.24c.



(a) *Point cloud* hasil pemindaian



(b) Heightmap dari *pointcloud*.



(c) Lingkungan dimunculkan.

**Gambar 3.24:** Proses penataan objek dari data *point cloud*

### 3.13.1 Metode *Downsampling*

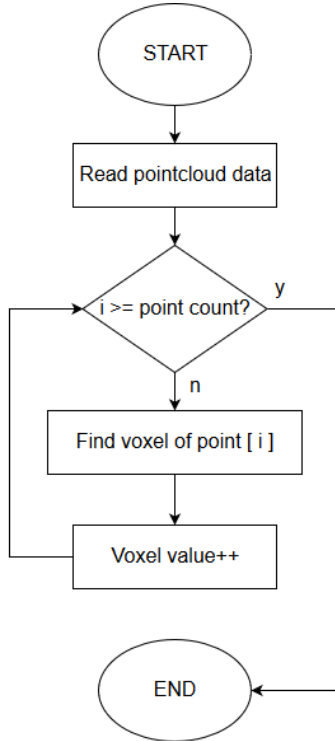
Metode yang digunakan dalam melakukan *downsampling* adalah *voxel grid filter*. Untuk melakukan *voxel grid filtering*, digunakan sebuah matriks tiga dimensi yang merepresentasikan sebuah *voxel grid* yang ukuran tiap *voxel*-nya ( $dx, dy, dz$ ) sama. Tiap *voxel* akan merepresentasikan keberadaan titik-titik di dalam *voxel* itu. Ukuran tiap *voxel* ditentukan sebesar 0,25 m. Karena satuan yang



digunakan di Unity adalah meter, maka digunakan Persamaan 3.1 untuk mendefinisikan ukuran  $grid[i, j, k]$ .

$$\begin{pmatrix} i \\ j \\ k \end{pmatrix} = \begin{pmatrix} (P_{max}.x - P_{min}.x)/dx \\ (P_{max}.y - P_{min}.y)/dy \\ (P_{max}.z - P_{min}.z)/dz \end{pmatrix} \quad (3.1)$$

Dimana  $P_{max}$  adalah koordinat yang terbentuk dari titik  $x, y, z$  maksimum dan  $P_{min}$  adalah koordinat yang terbentuk dari titik  $x, y, z$  minimum yang didapatkan saat pemindaian.



**Gambar 3.25:** Flowchart proses downsampling.

Seluruh sel di *grid* tersebut diisi nilai nol. Kemudian, tiap titik ( $p$ ) dalam *point cloud* dicek untuk dicari di *voxel* manakah titik itu berada. Untuk mencari *voxel* target, digunakan Persamaan 3.2.

$$\begin{pmatrix} i \\ j \\ k \end{pmatrix} = \begin{pmatrix} (p.x - P_{min}.x)/dx \\ (p.y - P_{min}.y)/dy \\ (p.z - P_{min}.z)/dz \end{pmatrix} \quad (3.2)$$

Tiap titik akan menambah nilai ke *voxel* sebesar satu. *Flowchart* dari proses ini ditampilkan pada Gambar 3.25. Setelah proses pengecekan seluruh titik *point cloud* selesai, nilai tiap *voxel* dalam *grid* tersebut adalah jumlah titik yang terdapat dalam sel tersebut dan apabila nilai yang dimiliki *voxel* tersebut lebih tinggi sama dengan *threshold* maka sel tersebut akan mewakili keberadaan titik-titiknya. Begitu pula sebaliknya, apabila nilai *voxel* tersebut di bawah *threshold* maka sel tersebut tidak mewakili keberadaan titik-titiknya.

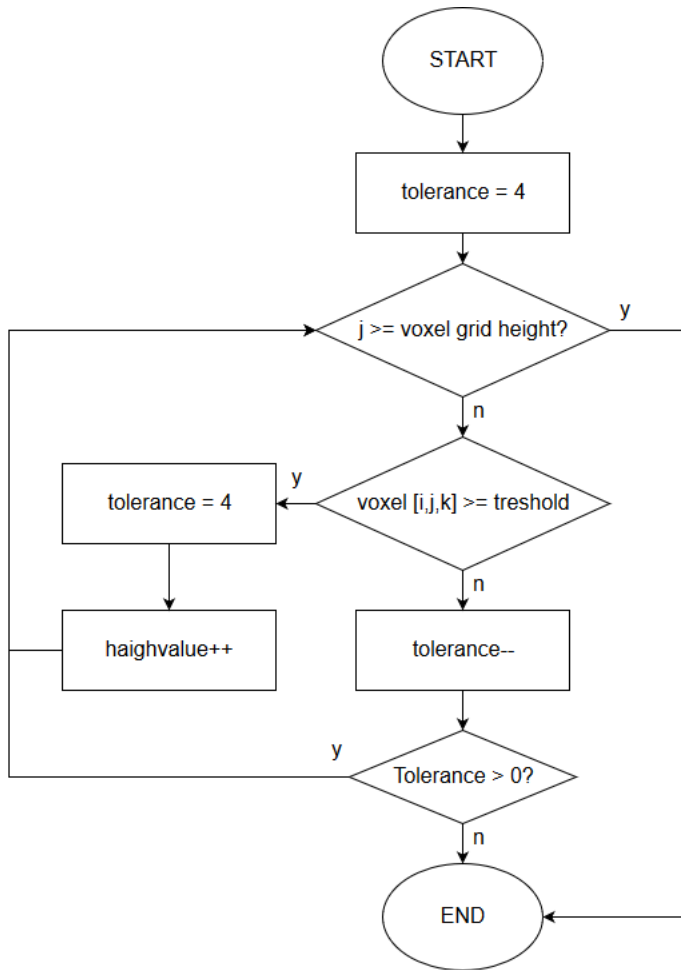
### 3.13.2 Algoritma *Heightmapping*

*Heightmapping* dilakukan dengan menggunakan sebuah matriks dua dimensi yang ukuran dimensi pertama ( $i$ ) dan keduanya ( $j$ ) sama dengan ukuran dimensi pertama ( $i$ ) dan ketiga ( $k$ ) milik matriks *grid voxel*. Tiap elemen matriks tersebut mewakili ketinggian di sebuah koordinat ( $i, k$ ) dalam *voxel grid*.

Pada dasarnya, untuk mendapatkan ketinggian dalam suatu koordinat maka dicari nilai  $y$  atau  $j$  tertinggi dalam koordinat tersebut. Tetapi dalam pembuatan aplikasi ini ada sebuah pertimbangan yaitu penggunaan aplikasi yang dapat digunakan di dalam ruangan. Apabila hanya menggunakan nilai  $y$  maksimum sebagai tinggi di suatu koordinat, maka besar kemungkinan akan mengam-bil posisi langit-langit ruangan. Oleh karena itu, seperti yang gambar tunjukkan, ada pengecekan toleransi sebelum memasukkan nilai ketinggian. Penggunaan nilai toleransi ditunjukkan pada Gambar 3.26.

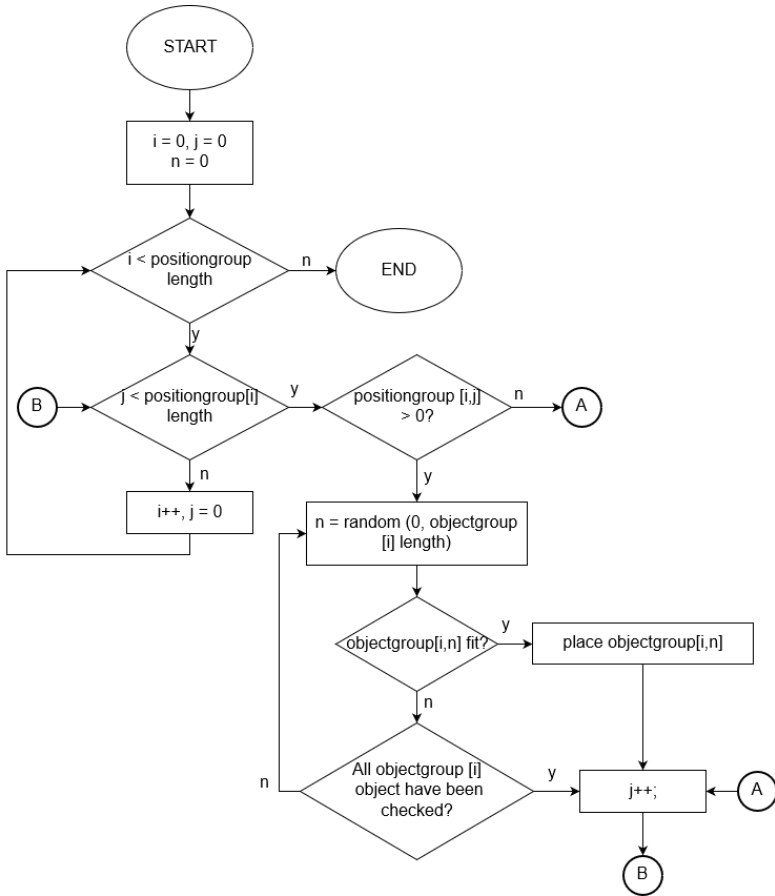
### 3.13.3 Algoritma Peletakan Objek *Virtual*

Objek-objek yang akan diletakkan sebelumnya dikelompokkan berdasarkan ketinggiannya. Selain itu, panjang dan lebar bagian



**Gambar 3.26:** *Flowchart proses heightmapping.*

objek yang menyentuh tanah juga didefinisikan. Sebagai contoh, panjang dan lebar dari sebuah pohon adalah panjang dan lebar batang bagian bawahnya. Berikut adalah tabel pengelompokan objek beserta ukuran tiap objeknya.



**Gambar 3.27:** Flowchart proses pemilihan objek yang akan diletakkan pada suatu koordinat.

Tiap koordinat dalam *heightmap* yang memiliki ketinggian lebih dari nol akan dikelompokkan berdasarkan ketinggiannya. Jumlah kelompok dan batas minimal ketinggian tiap kelompok koordinat mengikuti pengelompokan objek.

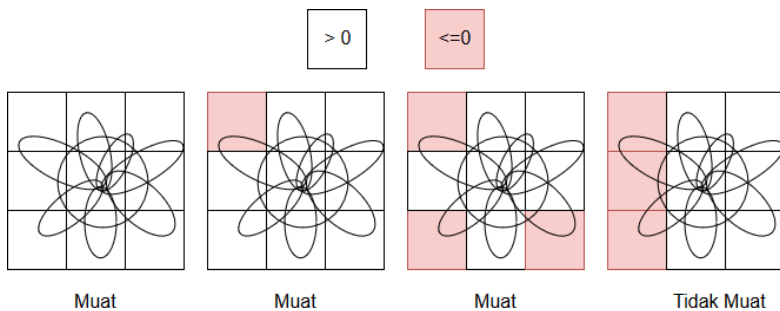
Setelah semua koordinat dikelompokkan, dimulai dari kelompok yang memiliki nilai ketinggian minimal tertinggi, tiap kelompok

**Tabel 3.3:** Tabel pengelompokan objek berdasarkan tinggi

Tinggi Minimum (m)	Nama Objek	Panjang (m)	Lebar (m)
1,5	Pohon Tinggi 1	1	1
	Pohon Tinggi 2	1	1
0,75	Batu 1	1,5	1,5
	Batu 2	1,5	1,5
	Pohon Rendah	0,75	0,75
0	Rumput	0	0

diiterasi tiap koordinat anggotanya untuk meletakkan objek pada koordinat tersebut.

Proses menentukan objek apa yang akan diletakkan pada suatu koordinat digambarkan pada Gambar 3.27. Tiap kelompok objek yang memiliki tinggi minimal yang sama dengan dengan tinggi minimal dari kelompok koordinat milik koordinat tersebut dipilih salah satu objek anggotanya secara acak.

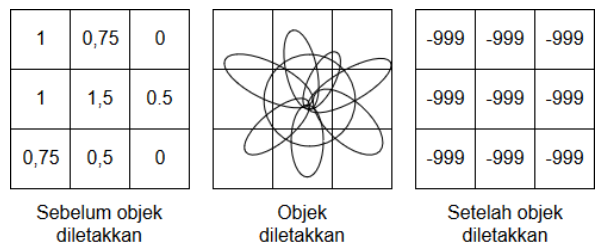


**Gambar 3.28:** Metode penentuan muat tidaknya objek pada suatu koordinat.

Sebelum meletakkan objek tersebut akan dicek apakah objek tersebut muat dalam koordinat tersebut. Kriteria yang membuat sebuah objek muat atau tidak dijelaskan dalam Gambar 3.28

Apabila objek tidak muat maka akan mengulangi ke proses pemilihan objek. Apabila objek muat, maka objek akan diletakkan

pada koordinat tersebut dengan rotasi sumbu  $y$  acak. Pada Gambar 3.28 ditunjukkan kotak putih dan merah, kotak merah menyatakan bahwa nilai di kotak tersebut kurang dari nol. Kotak nol merupakan area bersih yang akan digunakan pengguna untuk bereksplorasi yang apabila diisi objek, maka area eksplorasi pengguna semakin sedikit.



**Gambar 3.29:** Nilai tiap koordinat di sekitar objek sebelum dan setelah objek diletakkan.

Setelah objek diletakkan, seperti yang digambarkan pada Gambar 3.29, koordinat-koordinat yang terkena objek akan diberi nilai -999 untuk menandai bahwa koordinat tersebut tak dapat ditempati lagi untuk mencegah dua atau lebih objek saling bertabrakan.

### 3.13.4 Peletakan Pintu dan Manusia Purba *Virtual*

Khusus untuk objek pintu *virtual* dan manusia purba diletakkan secara khusus. Pintu *virtual* diletakkan di pusat lingkungan *virtual*. Apabila panjang ( $m$ ) *heightmap* lebih besar dari lebarnya ( $n$ ) maka manusia purba pertama diletakkan pada koordinat  $((n/2), (m/4))$  dan manusia purba kedua diletakkan pada koordinat  $((n/2), (3m/4))$ . Sebaliknya, apabila panjang ( $m$ ) *heightmap* lebih kecil dari lebarnya ( $n$ ) maka manusia purba pertama diletakkan pada koordinat  $((n/4), (m/2))$  dan manusia purba kedua diletakkan pada koordinat  $((3n/4), (3m/2))$ .

## BAB 4

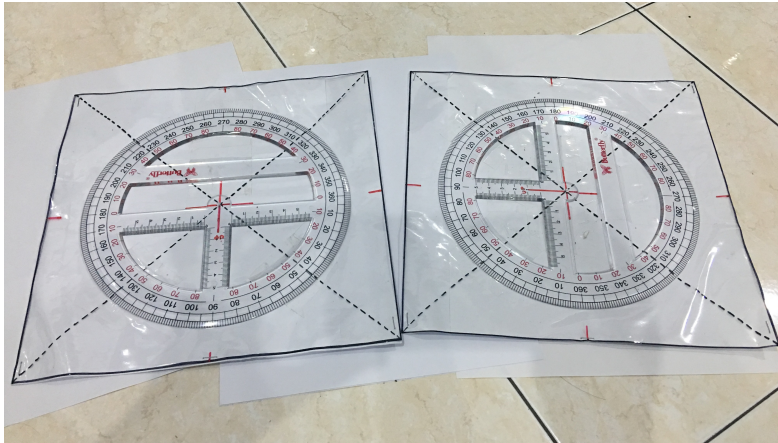
### PENGUJIAN DAN ANALISIS

Terdapat dua jenis penelitian yang dilakukan, yaitu pengujian performa sistem dan pengujian ketergunaan. Pengujian performa sistem dilakukan untuk menguji performa aplikasi saat digunakan dan pengujian ketergunaan dilakukan untuk mengetahui apakah tujuan yang ingin dicapai saat penggunaan tercapai.

#### 4.1 Pengujian Performasi Sistem

Pengujian performa pada dipecah menjadi dua jenis pengujian, yaitu pengujian *frame rate* dan sistem penataan lingkungan *virtual*. Pengujian ini dilakukan untuk mengetahui *frame rate* rata-rata saat pemindaian dan eksplorasi sekaligus untuk mengetahui jumlah rata-rata titik *point cloud* tiap *frame* saat pemindaian dan mengetahui batas luas penggunaan aplikasi yang nyaman digunakan. Proses-proses seperti pemindaian dan *rendering* dunia *virtual* dapat mempengaruhi *frame rate* dari aplikasi yang apabila terlalu besar maka *frame rate* dapat menjadi rendah atau umum disebut *FPS drop*. Apabila *FPS drop* sering terjadi, maka pengguna akan merasa kurang nyaman dalam proses pemindaian dan sistem *position tracking* juga dapat mengalami gangguan yang dapat menyebabkan hasil pemindaian menjadi rusak dan bergesernya dunia *virtual* yang ditampilkan.

Pengujian sistem penataan lingkungan *virtual* dilakukan untuk mengetahui waktu yang dibutuhkan untuk penataan lingkungan *virtual* dan akurasi penataan lingkungan *virtual*. Pengujian penataan lingkungan dilakukan di dalam sebuah ruangan berukuran 6,4 m x 12 m yang berisi sepuluh benda. Karena pengelompokan objek dibagi menjadi dua kelompok, kelompok benda rendah yang memiliki tinggi kurang dari 1,5 m dan lebih tinggi dari 0,75 m dan kelompok benda tinggi yang memiliki tinggi lebih dari 1,5 m maka pengujian dilakukan dalam tiga jenis ruangan. Ketiga jenis ruangan tersebut adalah ruangan yang berisi benda-benda rendah, ruangan yang berisi benda-benda tinggi dan kelompok yang berisi benda-benda dengan ketinggian campuran. Pengujian sistem penataan lingkungan *virtual* diawali dengan mendata tiap objek dalam suatu ruangan di dunia nyata. Kemudian, setelah dunia *virtual* ditampilkan, tiap



**Gambar 4.1:** Penanda posisi kubus.

objek di lingkungan dunia nyata dicek apakah objek nyata tersebut terwakilkan di dunia *virtual* atau tidak. Benda yang mewakili harus benda yang bergolongan tinggi sama dengan benda di dunia nyata, apabila tinggi benda nyata masuk dalam golongan rendah maka benda *virtual* yang ditampilkan harus benda rendah. Selain itu, pengujian ini juga mengambil data waktu yang diperlukan untuk menata lingkungan *virtual* dan translasi dan rotasi semesta *virtual* terhadap objek-objek di dunia nyata. Skenario pengujian adalah sebagai berikut.

1. Skenario pengujian yang dilakukan untuk sistem pemindaian adalah partisipan memulai proses pemindaian pada tengah ruangan, kemudian akan berkeliling menandai keseluruhan ruangan sampai selesai. Setelah pemindaian selesai, akan diambil data *frame rate* dan *point cloud* tiap *frame* pada tiga puluh detik pertama proses pemindaian.
2. Skenario pengujian untuk pengujian eksplorasi dunia *virtual* dilakukan dengan memasuki pintu virtual, kemudian partisipan diminta untuk mengelilingi dunia *virtual* dan akan diambil data *frame rate* selama tiga puluh detik pertama di dunia *virtual*.



3. Skenario pengujian penataan ruangan dilakukan setelah partisipan mencukupi pemindaian ruangan. Pengujian dilakukan dengan pengukuran translasi dan perputaran semesta dari lingkungan *virtual*. Pengukuran dilakukan menggunakan sebuah kubus di dunia *virtual* dan dua buah penanda yang ditunjukkan pada Gambar 4.1. Penanda pertama diletakkan pada posisi kubus sebelum proses penataan lingkungan. Setelah proses penataan selesai, penanda kedua diletakkan di posisi kubus. Peletakan kedua penanda tidak hanya menyamakan posisi tetapi juga arah rotasi dari kubus. Selisih jarak dan rotasi dari kedua penanda kemudian akan dicatat.

Perangkat yang digunakan dalam pengujian performansi adalah Apple iPhone 6S Plus yang memiliki spesifikasi teknis sebagai berikut

<b><i>Processor</i></b>	: Apple A9 Dual-core 1.84 GHz Twister
<b><i>RAM</i></b>	: 2 GB
<b><i>GPU</i></b>	: PowerVR GT7600 (six-core graphics)
<b><i>HDD</i></b>	: 64 GB
<b><i>Operating System</i></b>	: iOS 11.2

#### 4.1.1 Pengujian *Frame Rate*

*Frame rate* adalah frekuensi penggantian gambar yang ditampilkan di layar. *Frame rate* minimum suatu aplikasi agar nyaman digunakan adalah 30 FPS, dan *frame rate* yang dianjurkan adalah 60 FPS [18]. Hasil pengujian *frame rate* dari aplikasai adalah sebagai berikut.

Pengujian *frame rate* dilakukan dalam tiga area. Area pertama adalah sebuah ruangan buatan berukuran 5 m x 10 m yang berisi hanya sepuluh buah objek. Area kedua adalah ruangan Laboratorium B201 Teknik Komputer ITS yang berukuran 8 m x 6,5 m dan dan ruangan ini memiliki banyak objek di dalamnya. Area ketiga adalah lapangan voli Fakultas Teknologi Elektro ITS dengan ukuran 23 m x 10,5 m.

**Tabel 4.1:** Hasil pengujian sistem pemindaian pada area pertama

No.	<i>Point Cloud</i>			<i>Frame rate</i>		
	<i>Min</i>	<i>Max</i>	<i>Mean</i>	<i>Min</i>	<i>Max</i>	<i>Mean</i>
1	0	543	412.39	30	150	59.52
2	0	530	370.26	28	131	59.49
3	46	505	309.48	29	123	59.63
4	0	487	282.59	25	170	59.66
5	0	472	225.56	26	182	59.60
6	0	560	399.43	28	245	59.68
7	0	538	390.71	21	180	59.60
8	0	808	285.08	26	241	59.64
9	0	530	378.60	18	191	59.67
10	0	532	400.69	30	148	59.50

## Pengujian Area Pertama

Pada Tabel 4.1 ditunjukkan hasil pengujian tersebut dan *frame rate* rata-rata yang didapatkan pada tiap pengujian lebih dari *frame rate* minimum, 30 FPS. Pada beberapa waktu, *frame rate* dapat lebih rendah dari *frame rate* minimum, hal ini disebabkan oleh proses penyimpanan yang cukup berat apabila *point cloud* yang didapatkan banyak. Seperti yang ditunjukkan pada tabel pengujian pemindaian, jumlah minimum *point cloud* pada beberapa pengujian menunjukkan nol. Hal ini disebabkan karena pada *frame* tersebut sistem ARKit tidak dapat menemukan fitur objek dalam gambar yang ditangkap.

Data jumlah *vertex* dari lingkungan *virtual* yang dimunculkan dan *frame rate* dalam pengujian sistem eksplorasi ditampilkan dalam tabel 4.2. Dari hasil pengujian didapatkan rata-rata *frame rate* dari setiap percobaan yang lebih tinggi dari *frame rate* minimum (30 FPS) dan *Frame rate*. Nilai *frame rate* rata-rata yang didapatkan lebih kecil daripada saat proses pemindaian karena perangkat perlu *me-render* jumlah *vertex* yang berjumlah ratusan ribuan di setiap pengujiannya, dengan rata-rata jumlah *vertex* sebesar 195.457 *vertex*.

**Tabel 4.2:** Hasil pengujian sistem eksplorasi pada area pertama

No.	Jumlah Total <i>Vertex</i>	<i>Frame rate (FPS)</i>		
		<i>Min</i>	<i>Max</i>	<i>Mean</i>
1	194678	29	66	59.38
2	194310	23	66	58.92
3	195400	17	100	59.40
4	195904	24	104	59.38
5	196612	29	80	59.36
6	193804	18	150	59.39
7	196094	30	66	59.41
8	195398	28	86	59.28
9	196082	18	73	59.39
10	196290	29	84	59.43

**Tabel 4.3:** Hasil pengujian sistem pemindaian lingkungan pada area kedua

No.	<i>Point Cloud</i>			<i>Frame rate</i>		
	<i>Min</i>	<i>Max</i>	<i>Mean</i>	<i>Min</i>	<i>Max</i>	<i>Mean</i>
1	16	437	190.73	28	114	59.44
2	0	463	171.09	30	112	59.53
3	0	364	172.15	32	112	59.54
4	0	396	177.86	24	101	59.46
5	0	382	176.64	29	173	59.46
6	0	430	201.36	30	150	59.59
7	0	381	153.32	28	81	59.48
8	0	371	154.76	32	94	59.50
9	0	355	123.01	48	80	59.52
10	0	317	134.55	49	79	59.51

## Pengujian Area Kedua

Pada area kedua, proses pemindaian juga menghasilkan *frame rate* rata-rata yang melampaui *frame rate* minimum pada setiap percobaannya seperti yang ditunjukkan oleh Tabel 4.3. *Frame rate* rata-rata dari tiap pengujian juga hampir mencapai *frame rate* ideal 60 FPS dan tidak jauh dari rata-rata ruangan sebelumnya walau-

**Tabel 4.4:** Hasil pengujian sistem eksplorasi lingkungan pada area kedua

No.	Jumlah Total <i>Vertex</i>	<i>Frame rate (FPS)</i>		
		<i>Min</i>	<i>Max</i>	<i>Mean</i>
1	208816	22	69	59.20
2	205296	29	63	59.49
3	206342	29	67	59.34
4	204420	22	80	59.33
5	198612	29	65	59.41
6	203066	23	76	59.39
7	200292	27	80	59.43
8	203692	15	73	59.21
9	205044	29	78	59.38
10	203066	26	86	59.39

pun ukuran ruangan yang lebih besar dan dengan objek-objek yang lebih banyak.

*Frame rate* rata-rata dari tiap percobaan eksplorasi pada area kedua memberikan nilai rata-rata (59,50 FPS) yang lebih besar dari area pertama (59,33 FPS). Hal ini mungkin saja terjadi karena jumlah *vertex* yang ditampilkan tidak besar dan juga perbedaan ukuran ruangan yang tidak jauh.

## Pengujian Area Ketiga

Area ketiga yang memiliki luas yang jauh lebih luas dibandingkan area-area sebelumnya memberikan hasil rata-rata *frame rate* sebesar 57,70 FPS, lebih rendah dari pengujian-pengujian sebelumnya. Hal ini dapat disebabkan oleh sistem penyimpanan yang harus menyimpan *point cloud* yang lebih banyak.

Sistem eksplorasi juga memberikan rata-rata *frame rate* yang lebih rendah dari kedua area lainnya, yaitu sebesar 58,01 FPS. Hal ini disebabkan oleh jumlah objek yang ditampilkan lebih banyak. Walaupun perbedaan dari rata-rata yang tidak terlalu jauh, tetapi dalam percobaan, diketahui sering terjadi *lagging* pada saat eksplorasi.

**Tabel 4.5:** Hasil pengujian sistem pemindaian lingkungan pada area ketiga

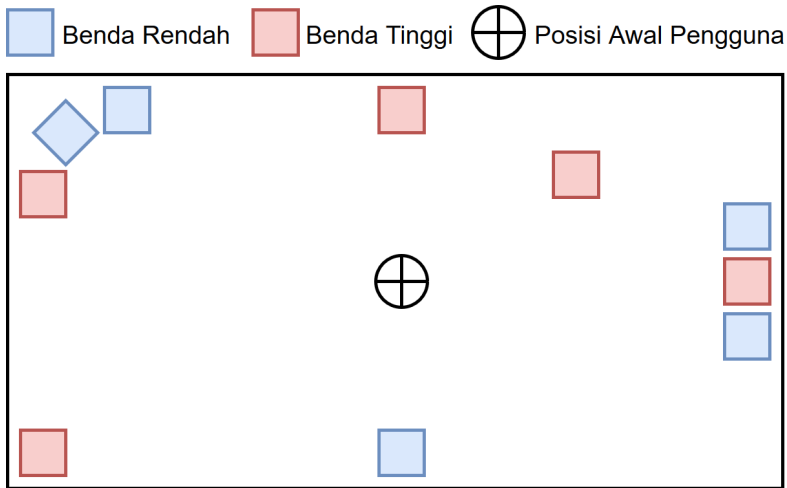
No.	<i>Point Cloud</i>			<i>Frame rate</i>		
	<i>Min</i>	<i>Max</i>	<i>Mean</i>	<i>Min</i>	<i>Max</i>	<i>Mean</i>
1	0	514	308.27	26	94	59.51
2	0	524	280.05	18	82	59.48
3	0	454	191.68	29	107	59.60
4	0	429	183.07	13	101	59.26
5	0	477	141.63	29	81	59.49
6	0	336	100.18	27	88	59.53
7	0	445	137.30	26	83	58.98
8	0	485	143.56	20	119	57.89
9	0	287	130.81	14	112	58.44
10	5	384	134.52	20	68	40.28

**Tabel 4.6:** Hasil pengujian sistem eksplorasi lingkungan pada area pertama

No.	Jumlah Total <i>Vertex</i>	<i>Frame rate (FPS)</i>		
		<i>Min</i>	<i>Max</i>	<i>Mean</i>
1	235084	11	98	56.35
2	244022	26	85	59.24
3	244004	21	93	55.09
4	244208	21	80	58.58
5	216962	27	72	58.54
6	225734	29	101	59.39
7	232226	29	61	58.20
8	248270	29	77	59.34
9	236432	0	131	59.18
10	241864	8	74	55.60

#### 4.1.2 Pengujian Sistem Penataan Lingkungan *Virtual*

Denah pengujian penataan dapat dilihat pada gambar 4.2. Pada penataan ruangan tersebut, ada beberapa objek yang diletakkan bersebelahan dengan objek lain. Objek-objek ini diletakkan berse-



**Gambar 4.2:** Denah Ruang Rendah.

belahan untuk menguji saat menggunakan ruangan campuran apakah aplikasi dapat membedakan benda-benda yang berketinggian berbeda walau posisinya bersebelahan.

Hasil pengujian penataan ruangan rendah dapat dilihat di Tabel 4.7. Dari sepuluh kali percobaan, diperoleh rata-rata persentase kesuksesan pemunculan benda sebesar 90% dan rata-rata waktu penataan sebesar 872,5 ms. Data kesuksesan tersebut hanya menunjukkan apakah benda tersebut terwakilkan atau tidak tanpa memedulikan translasi dan rotasi dari semesta *virtual*.

Hasil pengujian juga menunjukkan terjadi beberapa kali translasi. Apabila tanpa menghitung data *outlier*, yaitu pada percobaan lima, maka rata-rata translasi yang terjadi adalah sebesar 7,63 cm. Pada percobaan lima, selain terjadi translasi sebesar 221 cm, juga terjadi perputaran sebesar  $57^\circ$ . Translasi dan perputaran dapat terjadi apabila saat proses penataan berlangsung partisipan mengubah posisi dan rotasi perangkat. Hal tersebut terjadi karena proses penataan berlangsung dalam satu *frame* dan proses tersebut mengambil sumber daya dan waktu yang besar. Saat menunggu proses di *frame* tersebut selesai, ARKit tidak dapat melacak perpindah-

**Tabel 4.7:** Hasil pengujian sistem penataan lingkungan pada ruangan rendah

No.	Jumlah <i>Point Cloud</i>	Waktu (ms)	Persentase Keberhasilan	Translasi (cm)	Rotasi
1	4285	870	90%	4,8	0
2	4870	789	90%	12,5	0
3	4750	981	100%	2	0
4	4090	889	80%	11,8	0
5	5181	1005	80%	221	57
6	4125	440	90%	2	0
7	4142	892	90%	3	0
8	3603	916	90%	6,3	0
9	3771	899	100%	2	8
10	3527	1044	90%	24,3	0

an posisi dan rotasi dari perangkat. Hal ini diketahui dari data *profiler* salah satu percobaan yang menunjukkan bahwa *frame* saat proses penataan menghabiskan 1039 ms untuk komputasi seluruh *script* yang berjalan. Diantara seluruh *script* yang dijalankan saat itu adalah EnvironmentPlotter yang merupakan *script* yang bertugas untuk menata ruangan dan proses ini menghabiskan 77,2% dari keseluruhan waktu *script* atau 1020 ms.

Dari hasil sepuluh percobaan yang ditampilkan di Tabel 4.8, didapatkan rata-rata kesuksesan pemunculan benda sebesar 92% dan rata-rata ini lebih tinggi dari ruangan rendah. Waktu penataan rata-rata yang didapatkan lebih besar dari ruangan sebelumnya dengan waktu sebesar 835,4 ms.

Apabila mengabaikan data translasi di percobaan tiga dan enam yang merupakan data *outlier*, maka didapatkan rata-rata translasi sebesar 5,63 cm. Translasi ini lebih kecil dari ruangan benda rendah dan seperti ruangan benda rendah, terdapat beberapa percobaan yang menghasilkan translasi dan rotasi semesta yang besar.

Pada pengujian ini juga ditunjukkan pada percobaan satu dan sepuluh bahwa jumlah *point cloud* dari percobaan satu dua kali lipat

**Tabel 4.8:** Hasil pengujian sistem penataan lingkungan pada ruangan tinggi

No.	Jumlah <i>Point Cloud</i>	Waktu (ms)	Persentase Keberhasilan	Translasi (cm)	Rotasi
1	6415	445	100%	0	0
2	3990	615	90%	9,5	0
3	3910	844	100%	105	90
4	3638	849	80%	6	0
5	4020	879	80%	13,5	0
6	3573	851	100%	76	170
7	3464	960	90%	0	0
8	3662	1062	80%	8	0
9	3346	1020	100%	8	0
10	3219	829	100%	0	0

dari percobaan sepuluh tetapi percobaan satu memerlukan waktu yang lebih sedikit dari percobaan sepuluh.

**Tabel 4.9:** Hasil pengujian sistem penataan lingkungan pada ruangan campuran

No.	Jumlah <i>Point Cloud</i>	Waktu (ms)	Persentase Keberhasilan	Translasi (cm)	Rotasi
1	3955	831	80%	1,5	0
2	6987	880	80%	5	0
3	4048	918	90%	5	20
4	5171	911	80%	0,5	0
5	4678	944	90%	2	0
6	4122	908	80%	2	0
7	4083	913	70%	1	0
8	4209	983	100%	0,5	0
9	4303	865	90%	1	3
10	4501	941	90%	4	0



Pada pengujian sistem penataan di ruangan benda campuran, sistem penataan diuji apakah dapat dengan benar membedakan objek yang tinggi dan rendah. Pada Tabel 4.9, didapatkan rata-rata keberhasilan sebesar 85% dan rata-rata waktu penataan sebesar 909,4 ms. Data translasi rata-rata yang didapatkan adalah 2,25 cm dengan percobaan ketiga dan ke sembilan mengalami rotasi dimensi yang tidak sebesar rotasi-rotasi di perngjian-pengujian sebelumnya.

Data keberhasilan, translasi dan rotasi ini menunjukkan bahwa sistem penataan dapat dengan baik membedakan ketinggian objek walaupun objek-objek tersebut diletakkan di ruangan yang sama dan dengan posisi yang berdekatan.

Dari ketiga pengujian sistem penataan, diketahui sistem penataan memiliki persentase pemunculan benda yang tinggi sebesar 89%. Walaupun sistem penataan dapat merepresentasikan hampir semua objek nyata di dunia *virtual*, tetapi dalam pemunculannya ada translasi terhadap dunia nyata, translasi rata-rata keseluruhan yang didapatkan (mengabaikan data outlier) adalah 5,04 cm. Dari hasil pengujian penataan, didapatkan juga bahwa korelasi antara waktu generasi dan translasi memiliki korelasi yang lemah dengan koefisien sebesar 0,21. Dari tiga puluh percobaan panataan ruangan, terjadi rotasi semesta sebanyak enam kali.

## 4.2 Pengujian Ketergunaan

Pengujian ini dilakukan untuk mengukur ketergunaan atau *usability* yang merupakan sejauh mana suatu produk dapat digunakan oleh partisipan tertentu dalam mencapai suatu tujuan dengan efektifitas, efisiensi dan kepuasan dalam konteks yang dispesifikasikan [19].

Pengujian ini menggunakan tiga buah skenario utama, yaitu skenario menu utama, skenario pemindaian dan skenario eksplorasi. Tiap skenario memiliki beberapa tugas yang berbeda. Parameter yang diukur adalah waktu yang dihabiskan partisipan dalam menyelesaikan tiap tugasnya. Selain itu, juga diambil data kuesioner yang diberikan ke partisipan pada akhir percobaan.

Pengujian ketergunaan ini melibatkan tiga puluh orang yang diambil secara acak di Musum Sangiran, Kabupaten Sragen, Jawa Tengah. Alasan dari memilih Museum Sangiran sebagai lokasi

pengujian ketergunaan adalah untuk mencari partisipan yang berupa masyarakat umum dan telah mengetahui bentuk dan keadaan diorama nyata yang ditampilkan di dalam Museum Sangiran. Perangkat yang digunakan dalam pengujian ini adalah Apple iPhone 6s Plus yang sama dengan perangkat yang digunakan dalam pengujian performasi.

Dari ketiga skenario terdapat empat tugas yang harus diselesaikan oleh partisipan. Tugas pertama adalah saat awal aplikasi, partisipan diminta untuk memulai aplikasi dan proses pemindaian lingkungan. Kemudian tugas kedua adalah pemindaian lingkungan, partisipan diminta untuk memindai lingkungan sekitarnya hingga proses pemindaian selesai atau partisipan merasa cukup. Tugas ketiga adalah mencari pintu *virtual*, saat diorama *virtual* selesai dimunculkan partisipan akan diminta untuk mencari pintu *virtual*. Setelah partisipan memasuki pintu *virtual* partisipan diberikan tugas terakhir yaitu bereksplorasi di diorama *virtual* hingga partisipan merasa puas. Saat melakukan percobaan, partisipan akan didampingi untuk diberi tahu cara penggunaan aplikasi.

#### 4.2.1 Pengujian Efektivitas

Pengujian efektifitas dari tiap tugas dilakukan dengan menghitung *completion rate* dari tiap tugas, Hasil *completion rate* dapat dilihat pada Tabel 4.10.

**Tabel 4.10:** Hasil Pengujian *Completion Rate*.

No	Tugas	Dilakukan	<i>Completion Rate</i>
1	30	30	100%
2	30	29	96,66%
3	30	30	100%
4	30	29	96,66%

Pada tugas pemindaian dan eksplorasi ada sebuah percobaan yang gagal. Pada percobaan tersebut hasil pemindaian yang dilakukan partisipan tidak sesuai yaitu tidak terpindainya *ground* atau tanah di dunia nyata dan yang dianggap *ground* adalah benda la-

in yang ketinggiannya di atas tanah sehingga semesta *virtual* yang dimunculkan menjadi melayang dan tidak dapat dieksplorasi.

Selain completion rate, pengujian efektifitas juga dilakukan dengan menghitung waktu rata-rata yang dihabiskan partisipan dalam melakukan tugas-tugas yang diberikan. Saat partisipan melakukan eksplorasi para partisipan akan dibiarkan saja tanpa dibimbing untuk memberikan kebebasan atas seberapa lama partisipan dapat menikmati pengalaman eksplorasi hingga partisipan merasa puas. Dengan skenario seperti itu maka didapatkan hasil seperti yang ditunjukkan pada Tabel 4.11.

**Tabel 4.11:** Hasil Pengujian Waktu.

No	Tugas	Dilakukan	Waktu Rata-rata(s)
1	30	30	20,01
2	30	29	83,84
3	30	30	19,02
4	30	29	94,45

Dari data waktu yang didapatkan, partisipan masih membutuhkan waktu rata-rata selama 20,01 detik untuk memulai pemindaian, hal ini masih cukup lama dan dapat membuat pengguna aplikasi meninggalkan aplikasi karena kehilangan *flow*-nya [20]. Proses pemindaian dengan waktu rata-rata selama 83,34 detik masih terlalu lama karena masih lebih besar dari satu menit. Apabila dalam sebuah aplikasi pengguna perlu melakukan sebuah tugas sederhana dalam waktu lebih dari satu menit maka pengguna akan dapat meninggalkan aplikasi tersebut [20]. Waktu rata-rata yang dihabiskan partisipan dalam proses eksplorasi adalah 94,45 detik. Bila dibandingkan dengan waktu rata-rata yang dihabiskan dalam tahap-tahap sebelumnya sebesar 118,87 detik, waktu rata-rata eksplorasi masih lebih kecil. Hal ini menunjukkan bahwa partisipan masih harus menghabiskan waktu yang terlalu lama sebelum dapat menikmati proses eksplorasi.

## 4.2.2 Pengujian Kuesioner

Kuesioner pengujian dilakukan dengan memberikan lembar kuesioner setelah partisipan menyelesaikan tugas-tugas untuk pengujian efektifitas. Seperti yang dijelaskan sebelumnya, partisipan dari kuesioner adalah pengunjung Museum Manusia Purba Sangiran sebanyak tiga puluh orang dengan rentang usia lebih dari tujuh belas tahun. Rentang usia tersebut dipilih karena kuesioner perlu mengetahui pendapat tentang kemiripan dan pengamatan diorama asli dan diorama *virtual*.

Kuesioner yang diberikan kepada partisipan berisi enam belas pernyataan yang dapat dilihat pada Tabel 4.12 dan partisipan diberi lima opsi tingkat kesetujuan partisipan dengan pernyataan yang diberikan [21]. Berikut adalah opsi-opsi yang diberikan:

1. Sangat Tidak Setuju (STS)
2. Tidak Setuju (TS)
3. Netral (N)
4. Setuju (S)
5. Sangat Setuju (SS)

Pertanyaan pertama ditujukan untuk mengetahui bagaimana pengetahuan pengguna tentang teknologi *mixed reality*. Pertanyaan kedua dan ketiga ditujukan untuk mengetahui tampilan menu utama dan petunjuk-petunjuk yang diberikan sudah cukup jelas bagi partisipan.

Pertanyaan keempat ditujukan untuk mengetahui tingkat kemudahan proses pemindaian secara keseluruhan saat dioperasikan oleh partisipan. Pertanyaan kelima dan keenam ditujukan untuk mengetahui kemudahan dalam menemukan pintu *virtual* yang disajikan untuk memasuki diorama *virtual* dan terasa natural layaknya pintu di dunia nyata.

Pertanyaan tujuh dan delapan dimaksudkan untuk mengetahui kemiripan diorama *virtual* dengan diorama yang ditampilkan di dalam Museum Sangiran. Pertanyaan kesembilan sampai dengan kedua belas mewakili pernyataan partisipan tentang kenyamanan dan kemudahan dari kontrol eksplorasi dan untuk mengetahui apakah dengan kontrol yang disajikan eksplorasi diorama *virtual* terasa natural atau tidak. Pertanyaan ketiga belas ditujukan untuk mengetahui apakah penataan lingkungan *virtual* yang berdasarkan benda-benda di dunia nyata membantu partisipan untuk tidak me-

nabrak benda-benda di dunia nyata atau tidak. Pertanyaan keempat belas ditujukan untuk mengetahui kemudahan partisipan dalam menemukan pintu *virtual* setelah berkeliling di diorama *virtual*.

Pertanyaan kelima belas ditujukan untuk mengetahui pendapat partisipan tentang efektifitas manusia purba yang bergerak dalam diorama *virtual* dalam penyampaian materi kehidupan manusia purba. Pertanyaan keenam belas ditujukan untuk mengetahui apakah dengan diorama *virtual* yang menggunakan teknologi *Mixed Reality* dapat memberikan pengalaman baru ke partisipan atau tidak.

Hasil pengujian kuesioner ditunjukkan pada Tabel 4.13. Dengan menggunakan perhitungan skala Likert hanya 40% partisipan masih awam tentang *mixed reality*.

Sebanyak 63,33% partisipan beranggapan bahwa tampilan menu utama mudah dipahami. Petunjuk penggunaan juga diketahui mudah dipahami dengan 70% partisipan setuju dan ditambah dengan 30% yang sangat setuju. Pada saat melakukan pemindaian, tidak ada partisipan yang mengalami kesulitan dalam penggunaannya yang menunjukkan bahwa sistem pemindaian mudah untuk dioperasikan.

Setelah pintu *virtual* dimunculkan, tidak ada partisipan yang mengalami kesulitan dalam pencarian pintu *virtual* dengan 60% partisipan dapat menemukan pintu dengan mudah dan 50% partisipan setuju bahwa pintu *virtual* yang ditampilkan terasa natural layaknya pintu di dunia nyata.

Dari seluruh partisipan yang telah mengamati diorama koleksi Museum Sangiran, sebanyak 36,67% partisipan sangat setuju bahwa diorama *virtual* yang ditampilkan mirip dengan diorama yang ditampilkan di museum. Didapatkan 40% partisipan sangat setuju merasa bahwa objek-objek yang ditampilkan cukup detail.

Para partisipan merasa sistem eksplorasi tidak sulit dilakukan karena dari data yang didapatkan tidak ada partisipan yang mengalami kesulitan dengan 53,33% partisipan setuju. Sebanyak 53,33% partisipan yang setuju bahwa kontrol pergerakan di dunia *virtual* terasa natural. Dengan seluruh objek dunia *virtual* yang ditampilkan dan dengan tingkat detail yang cukup baik, animasi yang ditampilkan ke partisipan masih dirasa lancar oleh mayoritas partisipan dengan 43,33% partisipan setuju. Saat berada di dalam dio-

**Tabel 4.12:** Daftar pernyataan dalam kuisioner.

No	(Kode) Pernyataan
1	(MR) Anda mengetahui tentang teknologi <i>mixed reality</i> .
2	(Menu Utama) Tampilan menu utama mudah dipahami.
3	(Petunjuk Penggunaan) Petunjuk penggunaan mudah dipahami.
4	(Pemindaian) Pemindaian lingkungan mudah dioperasikan.
5	(Pencarian Pintu Awal) Pintu <i>virtual</i> mudah ditemukan sebelum memasuki diorama
6	(Kenaturalan Pintu) Pintu <i>virtual</i> terasa natural.
7	(Kemiripan Objek) Objek-objek di dalam lingkungan <i>virtual</i> yang ditampilkan mirip dengan objek-objek di dalam diorama museum.
8	(Detail Objek) Objek-objek yang ditampilkan cukup detail.
9	(Kemudahan Eksplorasi) Eksplorasi diorama <i>virtual</i> mudah dilakukan.
10	(Kontrol Pergerakan) Kontrol pergerakan terasa natural.
11	(Kelancaran Animasi) Animasi yang ditampilkan lancar.
12	(Sensasi Eksplorasi) Sensasi bergerak di dalam diorama <i>virtual</i> cukup meyakinkan.
13	(Pencegahan Tabrakan) Objek-objek di diorama <i>virtual</i> mencegah anda menabrak dengan benda-benda di lingkungan nyata selama eksplorasi diorama <i>virtual</i> .
14	(Pencarian Pintu Akhir) Pintu <i>virtual</i> mudah ditemukan setelah berkeliling di dalam diorama.
15	(Manusia Purba Bergerak) Manusia purba bergerak yang ditampilkan membantu dalam pemahaman kehidupan manusia purba.
16	(Pengalaman Baru) Eksplorasi <i>Mixed Reality</i> memberikan pengalaman baru dalam pembelajaran kehidupan manusia purba dibandingkan dengan diorama museum.

**Tabel 4.13:** Persentase respon partisipan terhadap pernyataan pada kuesioner

No.	Kode Pernyataan	Jawaban				
		STS	TS	N	S	SS
1	MR	10.00%	13.33%	40.00%	20.00%	16.67%
2	Menu Utama	0.00%	0.00%	3.33%	63.33%	33.33%
3	Petunjuk Penggunaan	0.00%	0.00%	0.00%	70.00%	30.00%
4	Pemindaian	0.00%	0.00%	6.67%	60.00%	33.33%
5	Pencarian Pintu Awal	0.00%	0.00%	3.33%	60.00%	36.67%
6	Kenaturalan Pintu	0.00%	0.00%	16.67%	50.00%	33.33%
7	Kemiripan Objek	0.00%	6.67%	20.00%	36.67%	36.67%
8	Detail Objek	0.00%	3.33%	40.00%	16.67%	40.00%
9	Kemudahan Eksplorasi	0.00%	0.00%	6.67%	53.33%	40.00%
10	Kontrol Pergerakan	0.00%	3.33%	10.00%	53.33%	33.33%
11	Kelancaran Animasi	0.00%	3.33%	13.33%	43.33%	40.00%
12	Sensasi Eksplorasi	0.00%	6.67%	16.67%	50.00%	26.67%
13	Pencegahan Tabrakan	0.00%	3.33%	30.00%	43.33%	23.33%
14	Pencarian Pintu Akhir	0.00%	3.33%	13.33%	50.00%	33.33%
15	Manusia Purba Bergerak	3.33%	3.33%	20.00%	36.67%	36.67%
16	Pengalaman Baru	3.33%	0.00%	16.67%	33.33%	46.67%

rama *virtual* sebanyak 50% partisipan setuju bahwa mereka merasa sensasi bergerak di dalam diorama *virtual* cukup meyakinkan saat mereka bereksplorasi.

Sistem penataan yang berdasarkan posisi benda di dunia nyata juga dianggap membantu dalam mencegah menabrak benda di dunia nyata oleh sebagian besar partisipan dengan 43,33% partisipan setuju. Sebanyak 50% partisipan setuju bahwa pintu *virtual* mudah ditemukan saat setelah bereksplorasi di diorama *virtual*. Hal ini membuktikan bahwa posisi pintu *virtual* dapat tetap berada pada posisi yang sama selama proses eksplorasi berlangsung.

Dengan manusia purba *virtual* yang ditampilkan bergerak, beberapa partisipan merasa bahwa gerakan yang ditampilkan belum membantu dalam pemahaman kehidupan manusia purba dengan 3,33% partisipan tidak setuju dan 3,33% partisipan merasa sangat tidak setuju. Akan tetapi, penampilan manusia bergerak di diorama *virtual* membantu sebagian besar partisipan dengan 36,67% partisipan setuju. Setelah mencoba teknologi *mixed reality* dalam pembelajaran kehidupan manusia purba, mayoritas partisipan beranggapan bahwa teknologi ini memberikan pengalaman baru dengan 46,64% partisipan yang sangat setuju.



## BAB 5

### PENUTUP

#### 5.1 Kesimpulan

Berdasarkan hasil implementasi dan pengujian diorama *virtual* menggunakan *mixed reality*, dapat ditarik beberapa kesimpulan sebagai berikut:

1. Aplikasi dapat digunakan hingga area berukuran 23 m x 10,5 m dengan *framerate* yang nyaman digunakan. Sistem penataan ruangan juga memiliki akurasi sebesar 89% walaupun terjadi translasi semesta rata-rata sebesar 5,04 cm dan rotasi semesta sebanyak enam kali dari tiga puluh percobaan.
2. Aplikasi mudah digunakan dengan 63,33% partisipan merasa bahwa menu utama mudah dipahami, 70% partisipan setuju bahwa petunjuk penggunaan mudah dipahami dan pemindaian dianggap mudah dioperasikan oleh 60% partisipan. Pintu *virtual* mudah ditemukan dengan 60% partisipan setuju dan eksplorasi diorama *virtual* juga mudah digunakan dengan 53,33% partisipan setuju. *Completion rate* dari proses pemindaian dan eksplorasi mendapatkan nilai sebesar 96,66%. Akan tetapi, waktu rata-rata yang dihabiskan partisipan dalam proses memulai pemindaian, pemindaian dan penataan masih terlalu lama karena masih lebih lama dari rata-rata waktu yang dihabiskan di proses eksplorasi.
3. Diorama *virtual* eksploratif dapat membantu penyampaian informasi kehidupan manusia purba dengan 36,67% partisipan setuju. Selain itu, 46,67% partisipan setuju eksplorasi *mixed reality* memberikan pengalaman baru dalam pembelajaran kehidupan manusia purba.

#### 5.2 Saran

Demi pengembangan lebih lanjut mengenai penelitian ini, disarankan beberapa langkah lanjutan sebagai berikut:

1. Menambah penyimpanan ruangan. Aplikasi belum memiliki kemampuan untuk menyimpan ruangan karena ketiadaan

kemampuan untuk membaca orientasi perangkat secara terus-menerus. Selain itu belum adanya kemampuan untuk mengenal posisi pengguna dalam ruangan yang tersimpan saat dimuat.

2. Mengembangkan dukungan untuk digunakan dengan *VR headset*. Saat ini belum ada alat yang dapat memberikan kamera perangkat kemampuan untuk mengambil gambar secara stereo agar tidak membuat pengguna merasa pusing saat stream kamera ditampilkan ke pengguna melalui *VR headset*.

## DAFTAR PUSTAKA

- [1] G. P. Rightmire and P. V. Tobias, “Homo erectus,” Oct 2017. (Dikutip pada halaman ix, 5, 6).
- [2] T. Ingicco, J. de Vos, and O. F. Huffman, “The oldest gibbon fossil (hylobatidae) from insular southeast asia: Evidence from trinit, (east java, indonesia), lower/middle pleistocene,” PLOS ONE, vol. 9, pp. 1–15, 06 2014. (Dikutip pada halaman 6).
- [3] J.-C. Piao and S.-D. Kim, “Adaptive monocular visualinertial slam for real-time augmented reality applications in mobile devices,” in Sensors, 2017. (Dikutip pada halaman ix, 7).
- [4] Apple Inc., “Arkit,” 2018. (Dikutip pada halaman 7).
- [5] Apple Inc., “About augmented reality and arkit,” 2018. (Dikutip pada halaman 7).
- [6] J. Gui, D. Gu, S. Wang, and H. Hu, “A review of visual inertial odometry from filtering and optimisation perspectives,” Advanced Robotics, vol. 29, no. 20, pp. 1289–1301, 2015. (Dikutip pada halaman 7).
- [7] National Instruments, “3D Imaging With NI Labview,” Jan 2017. (Dikutip pada halaman ix, 8).
- [8] R. Hartley, Multiple view geometry in computer vision. Cambridge, UK New York: Cambridge University Press, 2004. (Dikutip pada halaman 8).
- [9] M. Miesnieks, “Why is arkit better than the alternatives? 6d.ai medium,” Jul 2017. (Dikutip pada halaman 9).
- [10] P. Milgram and F. Kishino, “A taxonomy of mixed reality visual displays,” IEICE TRANSACTIONS on Information and Systems, vol. 77, no. 12, pp. 1321–1329, 1994. (Dikutip pada halaman ix, 9, 10).
- [11] B. Bray and M. Zeller, “What is mixed reality? - mixed reality,” Mar 2018. (Dikutip pada halaman ix, 10).

- [12] E. Adams, “The designer’s notebook: Postmodernism and the 3 types of immersion,” Jul 2004. (Dikutip pada halaman 11).
- [13] S. Bjork and J. Holopainen, “Patterns in game design (game development series),” 2004. (Dikutip pada halaman 11).
- [14] R. B. Rusu, “3d perception. 50% better. point cloud library,” 2010. (Dikutip pada halaman 11).
- [15] Z. Wang, H. Liu, Y. Qian, and T. Xu, “Real-time plane segmentation and obstacle detection of 3d point clouds for indoor scenes,” 10 2012. (Dikutip pada halaman ix, 12).
- [16] X.-F. Han, J. S. Jin, M.-J. Wang, W. Jiang, L. Gao, and L. Xiao, “A review of algorithms for filtering the 3d point cloud,” Signal Processing: Image Communication, vol. 57, pp. 103 – 112, 2017. (Dikutip pada halaman 12).
- [17] M. Afthon, “Eksplorasi maket virtual menggunakan augmented reality dan virtual reality,” Jun 2016. (Dikutip pada halaman 15).
- [18] Apple Inc., “Frame rate (ios and tvos),” Mar 2017. (Dikutip pada halaman 47).
- [19] H. Al-Kilidar, K. Cox, and B. Kitchenham, “The use and usefulness of the iso/iec 9126 quality standard,” Empirical Software Engineering, 2005. International Symposium on, pp. 7(pp, IEEE), 2005. (Dikutip pada halaman 55).
- [20] J. Nielsen, “Powers of 10: Time scales in user experience,” Oct 2009. (Dikutip pada halaman 57).
- [21] S. Jamieson, “Likert scales: How to (ab) use them,” vol. 38, pp. 1217–8, 01 2005. (Dikutip pada halaman 58).

## LAMPIRAN



(a) Foto diorama manusia purba Manusia Jawa.



(b) Foto diorama lingkungan purbakala.

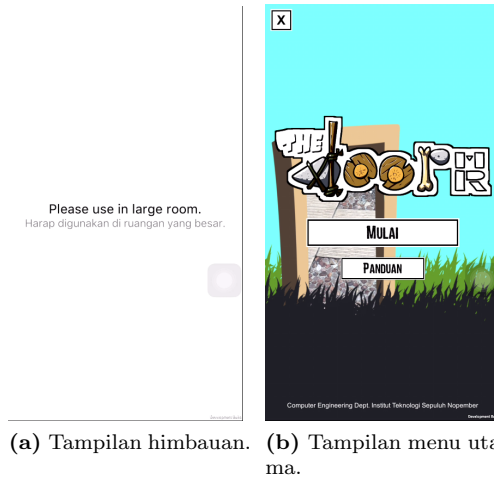
**Gambar 1:** Diorama manusia dan lingkungan purba Museum Manusia Purbakala Sangiran



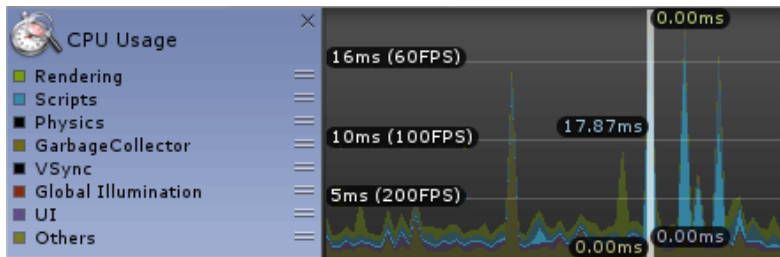
**Gambar 2:** Diorama manusia purba mengolah batu



**Gambar 3:** Pembatas di sekitar diorama agar tidak dimasuki



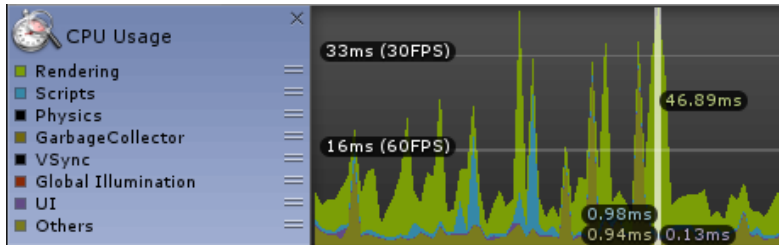
**Gambar 4:** Tampilan antar muka awal aplikasi



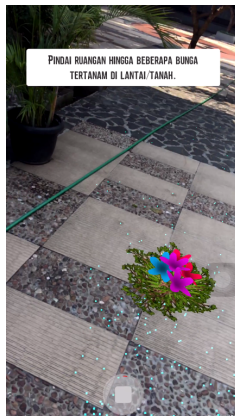
**Gambar 5:** Profiler proses pemindaian.

Overview	Total	Time ms
▼ Update.ScriptRunBehaviourUpdate	83.0%	18.08
▼ BehaviourUpdate	82.9%	18.08
DioramaManager.Update()	82.0%	17.87

**Gambar 6:** Status *script* DioramaManager.



**Gambar 7:** Profiler proses eksplorasi.



(a) Tampilan awal pemindaian.



(b) Tampilan konfirmasi hasil pemindaian.



(c) Tampilan *loading* penataan.

**Gambar 8:** Tampilan antar muka pemindaian





(a) Tampilan awal eksplorasi.



(b) Tampilan pintu *virtual*.



(c) Tampilan informasi dalam dunia *virtual*.

**Gambar 9:** Tampilan antar muka pemindaian



(a) Manusia purba 1.



(b) Manusia purba 2.

**Gambar 10:** Manusia purba *virtual* di dalam diorama *virtual*



(a) Benda tinggi

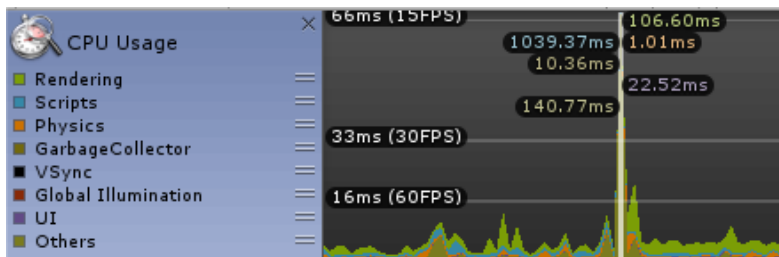


(b) Benda rendah

**Gambar 11:** Foto benda-benda yang digunakan dalam pengujian performansi



**Gambar 12:** Lokasi tes performasi



**Gambar 13:** Profiler proses penataan.

Overview	Total	Time ms
▼ Update.ScriptRunBehaviourUpdate	90.8%	1199.45
▼ BehaviourUpdate	90.8%	1199.45
▼ EventSystem.Update()	90.7%	1199.04
► EnvironmentPlotter.plotUpdate() [Coroutine: MoveNext]	77.2%	1020.45

**Gambar 14:** Status *Script* EnvironmentPlotter.

Tabel 1: Hasil pengujian sistem penataan lingkungan pada ruangan rendah														
No.	Jumlah <i>Point Cloud</i>	Waktu Generasi (ms)	Benda Rendah								Persentase Keberhasilan	Pergeseran (cm)		
			1	2	3	4	5	6	7	8			9	10
1	4285	870	o	o	o	o	o	o	o	o	o	x	90%	4,8
2	4870	789	o	o	o	o	o	o	o	o	o	x	90%	12,5
3	4750	981	o	o	o	o	o	o	o	o	o	o	100%	2
4	4090	889	x	o	o	o	o	o	o	o	o	x	80%	11,8
5	5181	1005	o	o	x	o	o	o	o	o	o	x	80%	221
6	4125	440	o	o	o	o	o	o	o	o	o	x	90%	2
7	4142	892	o	o	o	o	o	o	o	x	o	o	90%	3
8	3603	916	o	o	o	o	o	x	o	o	o	o	90%	6,3
9	3771	899	o	o	o	o	o	o	o	o	o	o	100%	2
10	3527	1044	o	o	o	o	o	o	o	o	o	x	90%	24,3

**Tabel 2:** Hasil pengujian sistem penataan lingkungan pada ruangan tinggi

No.	Jumlah <i>Point</i> <i>Cloud</i>	Waktu Generasi (ms)	Benda Tinggi										Persentase Keberhasilan	Pergeseran (cm)
			1	2	3	4	5	6	7	8	9	10		
1	6415	445	o	o	o	o	o	o	o	o	o	o	100%	0
2	3990	615	x	o	o	o	o	o	o	o	o	o	90%	9,5
3	3910	844	o	o	o	o	o	o	o	o	o	o	100%	105
4	3638	849	o	o	o	o	o	o	x	o	o	x	80%	6
5	4020	879	x	o	o	o	o	o	o	x	o	o	80%	13,5
6	3573	851	o	o	o	o	o	o	o	o	o	o	100%	76
7	3464	960	x	o	o	o	o	o	o	o	o	o	90%	0
8	3662	1062	x	o	o	o	o	x	o	o	o	o	80%	8
9	3346	1020	o	o	o	o	o	o	o	o	o	o	100%	8
10	3219	829	o	o	o	o	o	o	o	o	o	o	100%	0

**Tabel 3:** Hasil pengujian sistem penataan lingkungan pada ruangan campuran

No.	Jumlah <i>Point Cloud</i>	Waktu Generasi (ms)	Benda Rendah					Benda Tinggi					Persentase Keberhasilan	Pergeseran (cm)
			1	2	3	4	5	1	2	3	4	5		
1	3955	831	o	o	x	o	o	o	x	o	o	o	80%	1,5
2	6987	880	x	o	o	x	o	o	o	o	o	o	80%	5
3	4048	918	o	o	x	o	o	o	o	o	o	o	90%	5
4	5171	911	o	x	o	o	o	o	o	x	o	o	80%	0,5
5	4678	944	x	o	o	o	o	o	o	o	o	o	90%	2
6	4122	908	o	o	x	o	o	o	o	x	o	o	80%	2
7	4083	913	x	o	x	o	x	o	o	o	o	o	70%	1
8	4209	983	o	o	o	o	o	o	o	o	o	o	100%	0,5
9	4303	865	x	o	o	o	o	o	o	o	o	o	90%	1
10	4501	941	o	o	x	o	o	o	o	o	o	o	90%	4





(a)



(b)

**Gambar 15:** Foto lokasi pengujian Museum Sangiran



(a)



(b)



(c)



(d)

**Gambar 16:** Foto beberapa partisipan pengujian di Museum Sangiran



**Tabel 4:** Rekapitulasi respon partisipan terhadap pernyataan pada ku-  
esioner

No.	Kode Pernyataan	Jawaban				
		STS	TS	N	S	SS
1	MR	3	4	12	6	5
2	Menu Utama	0	0	1	19	10
3	Petunjuk Penggunaan	0	0	0	21	9
4	Pemindaian	0	0	2	18	10
5	Pencarioan Pintu Awal	0	0	1	18	11
6	Kenaturalan Pintu	0	0	5	15	10
7	Kemiripan Objek	0	2	6	11	11
8	Detail Objek	0	1	12	5	12
9	Kemudahan Eksplorasi	0	0	2	16	12
10	Kontrol Pergerakan	0	1	3	16	10
11	Kelancaran Animasi	0	1	4	13	12
12	Sensasi Eksplorasi	0	2	5	15	8
13	Pencegahan Tabrakan	0	1	9	13	7
14	Pencarian Pintu Akhir	0	1	4	15	10
15	Manusia Purba Bergerak	1	1	6	11	11
16	Pengalaman Baru	1	0	5	10	14

*Halaman ini sengaja dikosongkan*

## BIOGRAFI PENULIS



Bima Panji Yudasmara, lahir pada 05 November 1997 di Purworejo, Jawa Tengah. Penulis menempuh pendidikan S1 Departemen Teknik Komputer bidang studi *Game and Mobile Apps* Fakultas Teknologi Elektro ITS. Penulis aktif pada UKM IFLS ITS pada tahun 2014-2015 dan menjadi panitia dalam INOCHI ITS pada tahun 2015. Penulis merupakan anggota Game Developer Arek Suraboyo (GADAS) sejak tahun 2015. Pada tahun 2015-2016 penulis menjabat sebagai Koordinator Perlengkapan pada acara tahunan Global Game Jam Surabaya hingga pada tahun 2016-2017 penulis menjabat sebagai Ketua Pelaksana acara yang sama. Sejak tahun 2016 penulis merupakan anggota aktif Lab-B201Crew dan menjabat sebagai Koordinator Asisten Laboratorium pada tahun 2017-2018. Penulis juga mendirikan sebuah *startup game studio* bernama Calcatz pada tahun 2015. Penulis menghabiskan banyak waktu untuk mendalami pengembangan *game*, *game design*, pengembangan aplikasi android, *art*, dan topik mengenai VR, AR dan MR.

*Halaman ini sengaja dikosongkan*